BARRIER COVERAGE: DEPLOYING ROBOT GUARDS TO PREVENT INTRUSION

BY

STEPHEN KLODER

B.S., Georgia Institute of Technology, 2001
M.S., University of Illinois at Urbana-Champaign, 2004

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2008

Urbana, Illinois

Doctoral Committee:

Professor Seth Hutchinson, Chair
Associate Professor Jeff Erickson
Assistant Professor Indranil Gupta
Associate Professor Steven LaValle

Abstract

In this work, we formalize the problem of barrier coverage, that is, the problem of preventing undetected intrusion in a particular region using robot sensors. We give methods for complete barrier coverage in a two-dimensional polygonally-bounded region, for variable bounded-range line-of-sight sensors. To do this, we define barrier candidates, which allow us to search a limited set of possible guard configurations to find the minimum barrier, and convert the problem to a network flows problem. We extend this result to fixed-range line-of-sight, and fixed-range omnidirectional sensors. As the methods for fixed-range sensors are intractible, we also give efficient approximate methods. These approximate methods also use barrier candidates to quickly find good candidate deployments.

We use these barriers together with noncooperative zero-sum game theory to construct partial barriers. These are strategies for minimizing undetected intrusion when there is a limitation on available guard resources. We give equilibrium strategies for guards and intruder for the above three guard types in two dimensions. For variable-length sensors we derive strategies using barrier candidates. For fixed length guards we derive strategies using minimum fixed-length barriers, in conjunction with thick paths.

Acknowledgements

I would like to thank my family, who has always supported me in all my endeavors.

I would like to thank my advisor, Seth Hutchinson, who has taught me a great deal about shaping and presenting research. I would also like to thank the members of my committee, whose different perspectives greatly enhanced my research experience.

I would like to thank James Davidson, Sourabh Bhattacharya, and Sal Candido, with whom I have many invaluable exchanges of ideas and inspiration throughout our time as graduate students. I would also like to thank Kenton McHenry for his tireless efforts to maintain our lab's infrastructure.

Table of Contents

List of Figures

Chapter 1

Introduction

In this work, we solve a variety of problems related to *barrier coverage*. Barrier coverage is the problem of placing robot sensors to act as *guards* to protect a region from being entered by an *intruder*. This problem can be broadly divided into two classes of problems. The first is the problem of creating a minimum deployment of guards that successfully prevents undetected intrusion. We call such a deployment a *minimum complete barrier*. The second is the problem of deploying guards to minimize the probability of undetected intrusion. We call this the *partial barrier problem*. This second problem is particularly appropriate when the first problem has no solution, due to guard constraints. We will solve these problems for a variety of types of environments and guards.

Barrier coverage has a variety of practical applications. In building security [14, 45], a barrier can be a network of sensors to protect sensitive or valuable areas from unauthorized entry. In military settings, a barrier can be a security perimeter, around bases or between enemies. In robot herding [4, 51], a barrier can be an arrangement of virtual fences that keeps two herds apart, or that protects a herd from dangerous areas.

Barrier coverage appears in three different fields: coverage, sensor networks, and computational geometry. In robot coverage literature, barrier coverage appears as one of the three general types of coverage [17]. The other two are *blanket coverage* and *sweep coverage*. The goal of blanket coverage [19, 37, 41] is to ensure that every point in a region is seen by some stationary robot sensor. The goal of sweep coverage [8, 32, 57] is to ensure that every point in a region is seen by some moving robot sensor at least once. This problem can also be a single-robot problem [7,55,56]. Barrier coverage differs from the other two types in that its goal is covering every intrusion path, instead of covering every point.

In sensor network literature, barrier coverage appears as moat construction, or intruder detection. This is the problem of arranging a set of sensors across a region to detect an intruder trying to traverse the region. This set of sensors can be viewed as a barrier, and it can be generated using a variety of methods. For example, [21] uses potential fields, [11] uses incremental random deployments, and [31] generates a grid of sensors. In these examples, the barriers are usually generated for rectangular regions that have been preselected as moats around the protected area. The exception is in examples like [31], where the methods for rectangular moats can be extended to annulus-shaped moats.

In computational geometry, barrier coverage appears as one of the separation problems[1], specifically the problem of separating two point sets with the minimum number of shapes of a prespecified type. Some examples of these shapes are line segments [36], circles [3], wedges [20], or strips [20]. In these examples, the shapes are used to separate points in open space. In addition, [13] gives a solution for separating points in a polygon using chords. In separation problem examples, the set of separating shapes acts as a barrier, protecting the first set from an intruder who starts somewhere in the second set.

The fields of coverage, sensor networks, and computational geometry give different ideas of barriers, and consequently, different approaches. We generalize the notion of barrier coverage by combining elements from the different approaches from each of the fields. We use geometric approaches to accommodate a greater variety

[1]There is another type of separation problem: separation through motion [49]. This is the problem of finding the minimum motion necessary to move two sets so their convex hulls are disjoint.

of environments, and use sensor networks for a greater variety of problem domains. This allows us to look at different sensors, different goals, and different environment layouts, all under the same concept of barrier coverage.

Barrier coverage defines a variety of problems, since the parameters that define the problem can vary greatly, and each variation constitutes an interesting problem in its own right. The world can be two or three dimensional, and may have different terrains. There are a variety of possible guard sensors, including both bounded and unbounded range, in one direction or many. There are a variety of reasonable guard constraints, e.g. the number of guards, or energy consumption. Furthermore, intruders may have motion constraints, that guards can exploit. Any combination of these elements, and possibly others, produces a different problem, which may require its own unique solution.

In this work, we address two specific barrier coverage problems: finding the minimum complete barrier, and finding an optimal partial barrier.

- The *minimum complete barrier* is the guard deployment that completely prevents intrusion, while using the minimum guard resources.

- In situations where there are not enough guards to form a complete barrier, the *optimal partial barrier* is the optimal strategy for placing guards to maximize the probability of detecting an intruder. This involves also finding the intruder strategy that minimizes the probability of detection.

We solve these two problems in polygonal environments with polygonal holes for three types of guards.

- *Variable-length segments* are unidirectional line-of-sight sensors that can see up to a range that is specified by the deployer. The cost of a variable-length guard deployment is the total of all guards' ranges.

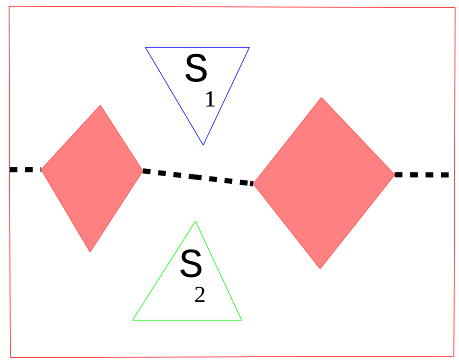


Figure 1.1: Minimum complete barrier for variable-range line-of-sight guards

- *Fixed-length segments* are unidirectional line-of-sight sensors that can see up to a preset range. The cost of a fixed-length segment guard deployment is the number of guards.

- *Fixed-range circles* are omnidirectional sensors that see up to a preset range. Like the fixed-length segment guards, the cost of a fixed-range circle guard deployment is the number of guards.

In Chapter 2 we give an efficient method for determining the minimum complete barrier for variable-length guards. This work was initially published in [28]. We demonstrate how to use network flows [23] to construct a minimum barrier in polynomial time. Figure 1.1 shows an example. The guards, with their sensor regions shown as thick dashed black lines, protect $\mathcal{S}_2$ from the intruder, who is known to start somewhere in $\mathcal{S}_1$.

In Chapter 3, we extend the variable-length results by giving barrier coverage methods for fixed-length segment guards. While we give exact solutions, we also show the problem is NP-complete. Therefore we also give approximate solutions, which are derived from the variable-length segment methods in Chapter 2.

In Chapter 4 we extend the fixed-length segments methods to find methods for fixed-range circle guards. Circles guards have all the capabilities of segment guards, but are more powerful, and require some additional considerations. We show that the minimum barrier problem is NP-Complete for circle guards, and give inefficient exact and efficient approximate methods derived from the segment guard methods in Chapter 3.

In Chapters 5 and 6, we extend the minimum complete barrier methods to find optimal partial barriers. We model this situation as a game, and apply game theory ideas from [2] to find optimal strategies for intruder and guards alike. Similar ideas of using game theory for automated security strategies appear in [40]. We show that an optimal partial barrier can be determined from the complete barrier. Chapter 5 gives partial coverage results for variable-length guards. This work appeared previously in [29]. Chapter 6 gives partial coverage results for both fixed-length segments and fixed-range circles.

Chapter 7 describes more long-term research goals connected to barrier coverage. We discuss them in order to give some insight into the long term prospects for the proposed research. These are problems that extend from the problems enumerated in the earlier chapters, but are more open-ended. This includes probabilistic guards, stochastic deployments, moving guards, intruders with knowledge and learning, and dealings with terrain and communication.

Chapter 2

Variable-length line-of-sight guards in two dimensions

In this chapter, we give an algorithm for finding the minimum complete barrier in the case of polygonal environments, point intruders, and variable-length one-directional line-of-sight guards.

2.1 Definition

The intruder is a point moving in the plane with coordinates $(x_I, y_I) \in \mathbb{R}^2$. The intruder can only be in the obstacle-free workspace $\mathcal{W} \subset \mathbb{R}^2$, which is connected, and bounded by polygons. The intruder is known to originate somewhere in the *start set*, $\mathcal{S}_1 \subset \mathcal{W}$, and attempts to travel to some point in the *stop set*, $\mathcal{S}_2 \subset \mathcal{W}$. Both $\mathcal{S}_1$ and $\mathcal{S}_2$ are compact and bounded by polygons. Figure 2.1 shows an example problem domain.

Each guard defines a line segment of variable length, and thus can be parameterized as $q_j = (x_j, y_j, \theta_j, r_j) \in \mathbb{R}^2 \times S^1 \times \mathbb{R}^+$, where (x_j, y_j) is the position of the guard, which can see in direction θ_j up to a distance of its range, r_j. Guards are further restricted in that they must reside in or along the workspace (i.e. $(x_j, y_j) \in \overline{\mathcal{W}}$, where $\overline{\mathcal{W}}$ is the closure of $\mathcal{W}$), and that they cannot see through walls, i.e. past points not in $\mathcal{W}$.

For each guard $q = (x, y, \theta, r)$ we define a *visibility region* $V(q)$ to be the set of points that the guard can see. Since a guard can see a straight line of length at most r up to an obstacle, $V(q)$ is the maximal connected component

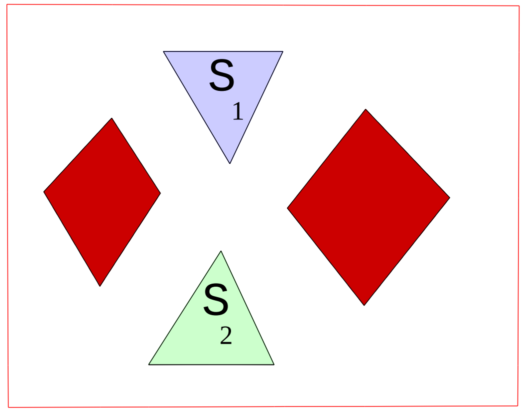


Figure 2.1: Sample barrier problem domain. $\mathcal{S}_1$ and $\mathcal{S}_2$ are, respectively, the intruder start set and stop set. The shaded regions are obstacles

of $\{(x + k\cos\theta, y + k\sin\theta) | \, 0 \leq k \leq r\} \cap \mathcal{W}$ that contains (x, y). These guards are called segment guards because the visibility region of each valid guard is a line segment. In this work we will often describe a segment guard in terms of its visibility region rather than its configuration.

A set of guards with configurations $\{q_1, \ldots, q_n\}$ is a *barrier* iff every path from $\mathcal{S}_1$ to $\mathcal{S}_2$ in $\mathcal{W}$ intersects $V(q_j)$ for at least one j. This means that the intruder cannot move from its start to its intended goal without being detected by a guard. Equivalently, $\{q_1, \ldots, q_n\}$ is a barrier iff $\mathcal{S}_1$ and $\mathcal{S}_2$ are in separate connected components of $\mathcal{W} - \bigcup\limits_{j=1}^{n} V(q_j)$.

Our goal is to find the minimum-length barrier, i.e. the set $\{q_1, \ldots, q_n\}$ that is a barrier, and has minimum $\sum\limits_{j=1}^{n} r_j$. This reflects situations where robots that see farther are more expensive to construct. Examples of this are long segment guards composed of many smaller segment guards, or fence systems.

2.2 Barrier candidates

In this section, we define *barrier candidates*. These are line segments in $\mathcal{W}$ that are relevant in constructing the minimum barrier. We will show that the minimum complete barrier must consist of barrier candidates. Barrier candidates have much in common with edges in visibility graphs [43].

If f_1 and f_2 are features in the environment (where a feature is an edge or a vertex), the shortest segment from a point in f_1 to a point in f_2 is called the *minimal segment* from f_1 to f_2. This segment is unique unless f_1 and f_2 are parallel edges. Not all minimal segments are useful in constructing minimum barriers. For example, if v is the vertex incident to edges e_1 and e_2 and the minimal segment from e_1 to e_3 has v as an endpoint, the segment is *redundant* unless it is also the minimal segment from e_2 to e_3. In Figure 2.2, the edges labeled a, b, c, and d are minimal segments. The segments b and c are redundant.

A segment s is *tangent* to a polygon at a vertex v if inside some neighborhood of v, the supporting line for s intersects the polygon at the boundary, but not the interior. A *bitangent* is a segment tangent to two polygon vertices. This bitangent is *separating* if the portions of the polygons in the vertices' neighborhoods are on opposite sides of the line through s, and *supporting* if they are on the same side. In Figure 2.2, e and h are tangents, f is a separating bitangent, and g is a supporting bitangent.

We refine our set of barrier candidates further by removing line segments that are definitely not useful in constructing a minimum barrier. We say that a segment is *admissible* if (1) its interior lies entirely inside $\mathcal{W}$, (2) it contains no points in the interior of $\mathcal{S}_1$ or $\mathcal{S}_2$, and (3) it is not redundant. The reasons for these restrictions will be explained below.

Definition: A line segment in $\mathcal{W}$ is a *barrier candidate* iff it is an admissible segment that is either

1. a minimal segment between obstacle edges,

2. a minimal segment from a vertex of $\mathcal{S}_i$ to an obstacle edge, and tangent to $\mathcal{S}_i$,

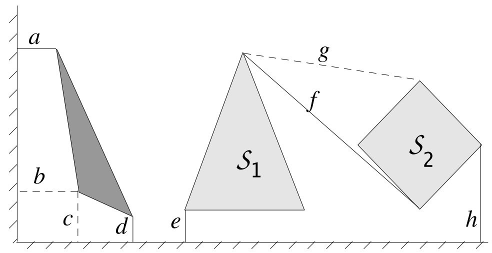


Figure 2.2: Sample barrier candidates. The dashed lines are not barrier candidates.

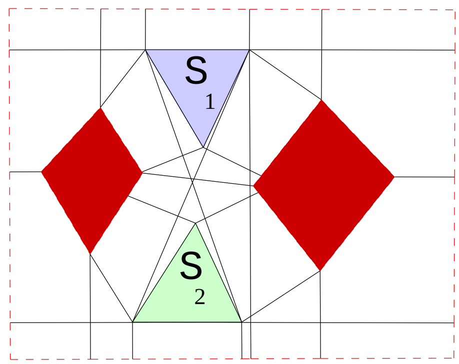


Figure 2.3: Barrier candidates for Figure 2.1. The boundary is dashed, and obstacles are solid dark red. The barrier candidates are solid.

$i = 1, 2,$

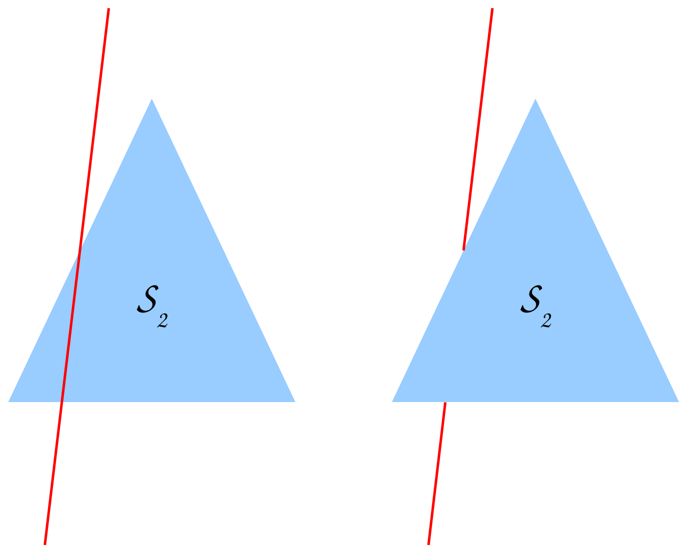
3. a separating bitangent between $\mathcal{S}_1$ and $\mathcal{S}_2$, or

4. a supporting bitangent between different vertices of $\mathcal{S}_1$, or between different vertices of $\mathcal{S}_2$,

Figure 2.3 shows all the barrier candidates for the domain from Figure 2.1.

Theorem 1: The minimum variable-length segment barrier separating $\mathcal{S}_1$ from $\mathcal{S}_2$ consists only of barrier candidates.

Proof: We will show that any barrier consists entirely of segments that are either barrier candidates, or can be shortened while maintaining the barrier. Since a minimum barrier cannot contain segments that can be shortened, it will not contain segments that are not barrier candidates.



(a) Before (b) After

Figure 2.4: Shortening by removing portions inside of $\mathcal{S}_i$.

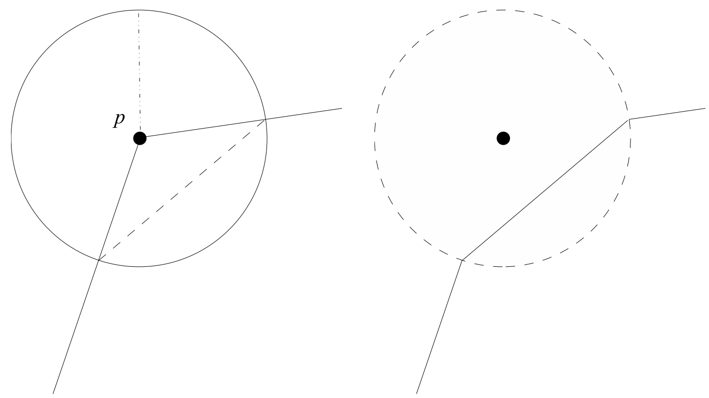
First, we show the necessity of requiring that segments be admissible to be barrier candidates. If a segment's interior does not lie entirely inside $\mathcal{W}$, portion(s) of the segment can be removed without affecting the barrier, effectively replacing the segment by multiple segments of shorter total length. If a segment intersects the interior of $\mathcal{S}_1$ or $\mathcal{S}_2$, that portion of the segment can be removed while preserving the barrier. See Figure 2.4. The requirement that a segment not be redundant will be proven below. Therefore, we need only consider admissible segments.

Next we show that a segment in a minimum barrier must be one of the five types enumerated in the definition of a barrier candidate. The segments that can be used to construct a barrier can be classified according to their endpoints. In general, a segment can have

1. one endpoint in the interior of $\mathcal{W} - \mathcal{S}_1 - \mathcal{S}_2$,

2. both endpoints on obstacle edges,

3. one endpoint at an obstacle edge, and one in $\mathcal{S}_1$ or $\mathcal{S}_2$, or



(a) Before (b) After

Figure 2.5: Shortening by replacing an internal vertex

4. both endpoints in $\mathcal{S}_1 \cup \mathcal{S}_2$.

We show for each of these cases, either the segment is a barrier candidate, or it can be shortened while preserving the barrier.

1. If the segment has an endpoint p in the interior of $\mathcal{W} - \mathcal{S}_1 - \mathcal{S}_2$, the barrier can always be shortened. The method of shortening depends on the degree of p, i.e. the number of segments incident to it.

   If the degree is one, the segment does not contribute to the barrier, and can be removed completely. If the degree is 2, select $\epsilon > 0$ such that the disk of radius ϵ centered at p lies entirely on the interior of $\mathcal{W} - \mathcal{S}_1 - \mathcal{S}_2$. Replace the portion of the barrier inside the disk with a line segment connecting the two points on the disk's boundary. By the triangle inequality, this barrier is shorter. See Figure 2.5. Note that this assumes the angle between segments at p is less than π. If it is equal to π, then the two segments can be viewed as a single segment, and p is no longer considered an internal vertex.

   If the degree is 3 or greater, consider the maximal regions of $\mathcal{W}$ separated by the complete barrier. Each region can be labeled by whether it contains points from $\mathcal{S}_1$, points from $\mathcal{S}_2$, or neither (if it contains points from both, it is not a

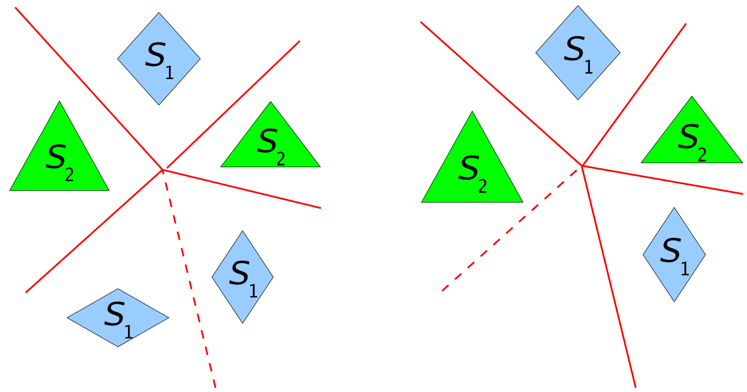
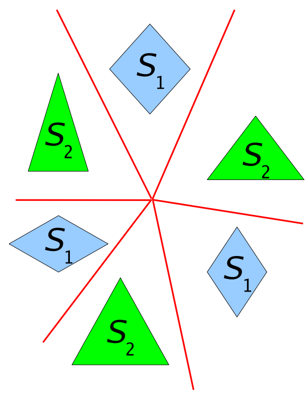
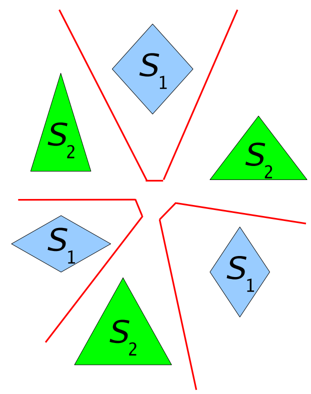


Figure 2.6: Shortening by removing an unnecessary segment. In both cases, the dashed segment can be removed



(a) Before



(b) After

Figure 2.7: Shortening by combining multiple regions.

barrier). Now, consider the regions that meet at this interior vertex. If there is a region with a "neither" label, or two adjacent regions with the same label, a segment can be removed. See Figure 2.6. Otherwise, pick two or more regions with the same label, and combine them with a method analogous to removing a degree-2 vertex. See Figure 2.7. The resulting barrier is shorter. Thus, no segment with an endpoint in the interior of $\mathcal{W} - \mathcal{S}_1 - \mathcal{S}_2$ can be included in a minimum barrier.

2. If a segment s connecting two obstacle edges is the shortest possible segment connecting those two edges, it is a minimal segment, and is a barrier candidate of type 1. If it is not the shortest, it can be shortened by moving one endpoint

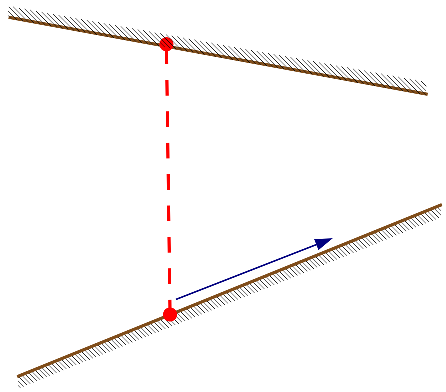


Figure 2.8: Shortening a non-minimal segment. The solid lines are obstacle segments, and the dotted line is a candidate barrier. Moving the endpoint along the obstacle boundary in the direction of the arrow shortens the barrier.

along the edge towards the minimal segment. See Figure 2.8. The same is true if s is redundant. For example, in Figure 2.2, c can be moved towards d. If s contains a point in $\mathcal{S}_1$ or $\mathcal{S}_2$, then it can be split into two separate segments that are dealt with in the next item. Therefore, if a minimum barrier contains a segment that connects two obstacle edges, it must be a minimal segment. This is a barrier candidate of type 1.

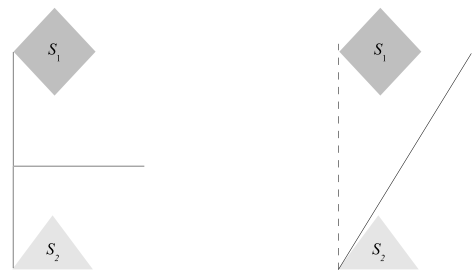
3. A segment s with an endpoint, v_1, on an obstacle edge and the other, v_2, in $\mathcal{S}_i$ can be shortened unless it is the shortest possible non-redundant segment between the obstacle edge and v_2. The method is the same as for the previous item. That v_2 must be on the boundary of $\mathcal{S}_i$ follows from the requirement that the segment be admissible. Furthermore, a segment connecting an obstacle to $\mathcal{S}_i$ cannot separate $\mathcal{S}_1$ from $\mathcal{S}_2$ on its own. It needs a second segment segment s' incident at v_2. If s is not tangent to $\mathcal{S}_i$, then s and s' combined can be shortened in a way analogous to removing an interior vertex. See Figure 2.9. Therefore, a segment of this type must be a minimal segment and tangent to $\mathcal{S}_i$ in order to be part of a minimum barrier. This is a barrier candidate of type 2.

(a) Before (b) After

Figure 2.9: Shortening a non-tangent

4. A segment connecting two points in $\mathcal{S}_1 \cup \mathcal{S}_2$ can be shortened unless it is a bitangent, using the method described in the previous item. If the segment is a supporting bitangent between $\mathcal{S}_1$ and $\mathcal{S}_2$, then it cannot separate on its own, and requires another segment. This new segment touches either an endpoint or the interior of the bitangent. If it touches the interior, it produces a degree-3 interior vertex, which is covered in item 1 above. If it touches an endpoint, the bitangent can be removed altogether. See Figure 2.10. If the segment is a separating bitangent between components of $\mathcal{S}_1$, or between components of $\mathcal{S}_2$, then it has no effect. This is because separating components of $\mathcal{S}_1$ does not change the set of locations that can be reached from $\mathcal{S}_1$; it only changes which locations an intruder can reach when starting at each component. Therefore such a bitangent can be removed. Therefore only separating bitangents between $\mathcal{S}_1$ and $\mathcal{S}_2$, supporting bitangents at $\mathcal{S}_1$, and supporting bitangents at $\mathcal{S}_2$ can be in minimum barriers. These are all barrier candidates (types 3 and 4).

If there exists a minimum barrier that contains segments that are not barrier candidates, then this barrier can be shortened using the methods described above. Therefore it is not minimum; this is a contradiction. Therefore, the minimum barrier



(a) (b)

Figure 2.10: Shortening a supporting bitangent: (a) is like Figure 2.6; in (b) the dashed segment can be removed without losing the barrier.

must consist exclusively of barrier candidates. □

Corollary 1: All vertices incident to two or more segments of the minimum barrier must be vertices of $\mathcal{S}_i$.

Proof: From the proof of Theorem 1, we know that all barrier candidates have endpoints at obstacles or $\mathcal{S}_i$. Therefore it suffices to show that any point p on an obstacle edge e is incident to at most one segment of the minimum barrier.

If p is in the interior of e, then it must be incident to a segment perpendicular to e; otherwise it would not be minimal. There is only one such line; there can be only one barrier candidate incident to p. If p is a vertex of e, where e meets obstacle edge e', then the other endpoints of the barrier candidates incident to p cannot be perpendicularly over e or e'. Otherwise they would not be minimal segments. Therefore, both segments must be in the unshaded region of Figure 2.11. However, if the barrier contains two such segments, it can be shortened as in Figure 2.5. This contradicts that it is the minimum barrier. Therefore, p cannot be incident to two or more segments of the minimum barrier. □

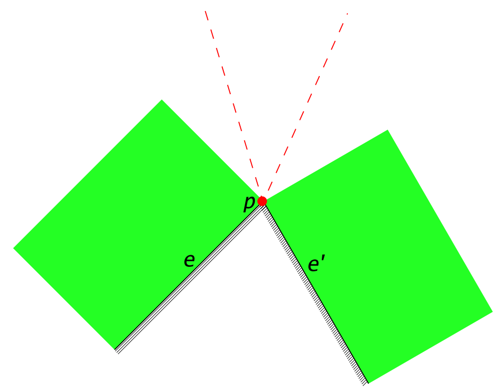


Figure 2.11: Barrier candidates incident to p cannot be inside the green shaded regions. The red dashed lines are examples of possible barrier candidates.

2.3 Barrier candidate graph and connectivity network

Given the set of barrier candidates, we construct a *barrier candidate graph* $\mathcal{G} = (V, E)$. The vertices V are all either (a) points where barrier candidates intersect each other or obstacles, or (b) obstacle vertices. The edges E are the pieces of barrier candidates or obstacle edges that connect these vertices. Every barrier candidate and every obstacle edge appears as a path through $\mathcal{G}$. This graph is finite since for any two features, there is a finite number of barrier candidates. The exception to this is when two obstacle edges can be parallel. In this case, it suffices to pick one perpendicular segment connecting the two obstacles. Section 2.4 shows why this is true.

Since $\mathcal{G}$ is planar, we can define a set F of faces for it. For every cycle in $\mathcal{G}$, we create a face iff the interior of the cycle's planar embedding is (a) minimal, i.e. does not overlap any other edges, and (b) entirely inside $\mathcal{W}$. Any minimal cycle not inside $\overline{\mathcal{W}}$ is an obstacle, and is not assigned a face.

From $\mathcal{G}$ we construct the dual graph [54] $\mathcal{G}^D = \left(V^D, E^D\right)$. Every vertex in V^D corresponds to a face in F, and every edge in E^D corresponds to an edge in E that separates faces in F, i.e. a portion of a barrier candidate. If two faces $f_i, f_j \in F$ share an edge $e_k \in E$, then the corresponding vertices $v'_i, v'_j \in V^D$ are connected by an edge $e'_k \in E^D$. Note that obstacles do not have corresponding vertices in V^D, as they have no corresponding faces in F.

This dual graph reflects the connectivity of the workspace. Traveling from one point in $\mathcal{W}$ to another point in a different region requires crossing some edges of $\mathcal{G}$. This is equivalent to following a path of corresponding dual edges in $\mathcal{G}^D$. Similarly, separating $\mathcal{S}_1$ from $\mathcal{S}_2$ requires finding an edge cut $C \subset E^D$ in $\mathcal{G}^D$ such that in the remaining graph $\mathcal{G}^D - C$ there is no path from a vertex inside a component of $\mathcal{S}_1$ to a vertex inside a component of $\mathcal{S}_2$. The minimum barrier is the edge cut C that separates $\mathcal{S}_1$ from $\mathcal{S}_2$ with the smallest total edge length of all e_k such that $e'_k \in C$.

We can find the min cut of $\mathcal{G}^D$ using a max-flow/min-cut algorithm [23]. Therefore, we construct the *connectivity network* $\mathcal{N}$ from $\mathcal{G}^D$ by adding (1) capacity values $c\left(e'_k\right)$ for every $e'_k \in E^D$, and (2) source and sink vertices v_{s_1} and v_{s_2} respectively. For each $e_k \in E$, $c\left(e'_k\right)$ is the length of e_k. Connect v_{s_1} to all the vertices corresponding to faces that intersect $\mathcal{S}_1$, and connect v_{s_2} to all the vertices corresponding to faces that intersect $\mathcal{S}_2$. Assign infinite capacity to all of these new edges to ensure that the minimum cut of $\mathcal{N}$ contains only edges in E^D. The same effect can be achieved by combining all vertices inside $\mathcal{S}_1$ into v_{s_1}, and the equivalent for v_{s_2}. An edge cut that separates v_{s_1} from v_{s_2} corresponds to a barrier that separates $\mathcal{S}_1$ from $\mathcal{S}_2$. Figure 2.12 shows the connectivity network derived from the barrier candidate graph for the example shown in Figure 2.3. The shaded regions are obstacles; they do not have corresponding vertices in $\mathcal{G}^D$ or $\mathcal{N}$, as they are not inside $\mathcal{W}$. Note that while $\mathcal{G}^D$ is necessarily planar, $\mathcal{N}$ may not be if there are multiple components of $\mathcal{S}_1$ or multiple components of $\mathcal{S}_2$.

The minimum edge cut separating v_{s_1} from v_{s_2} corresponds via the dual-graph relation to the shortest barrier that consists only of barrier candidate segments.

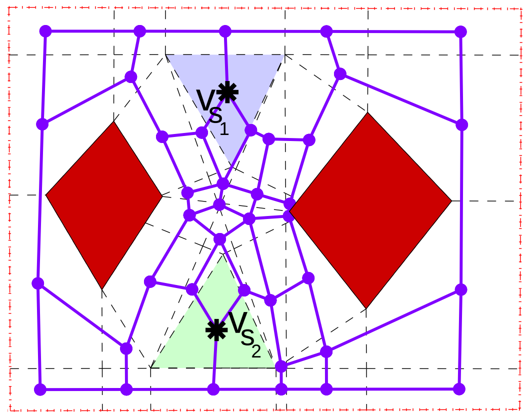


Figure 2.12: Connectivity network derived from barrier candidates in Figure 2.3. Solid edges are in the network, dashed edges are barrier candidates, and dotted lines are boundaries.

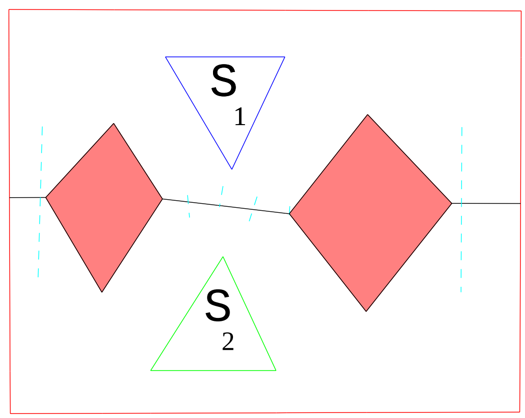
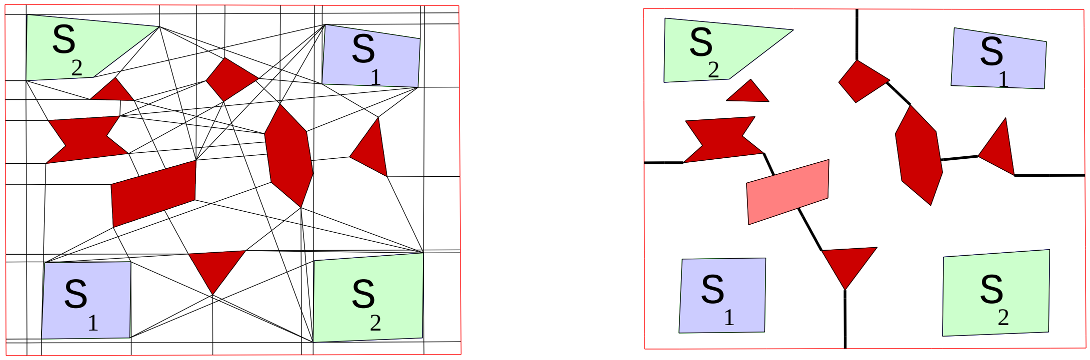


Figure 2.13: Minimum barrier. Dashed lines show corresponding edges in the dual graph

According to Theorem 1 there are no shorter barriers. Therefore this barrier is the minimum barrier of any type, and is the desired solution.

This minimum barrier can be found by solving the network flows min-cut problem, which is equivalent to the network flows max-flow problem [23]. This can be solved efficiently using augmenting paths [16] or preflows [25]. Figure 2.13 shows the minimum barrier derived from Figure 2.12. The dashed lines show the minimum cut for the connectivity graph.

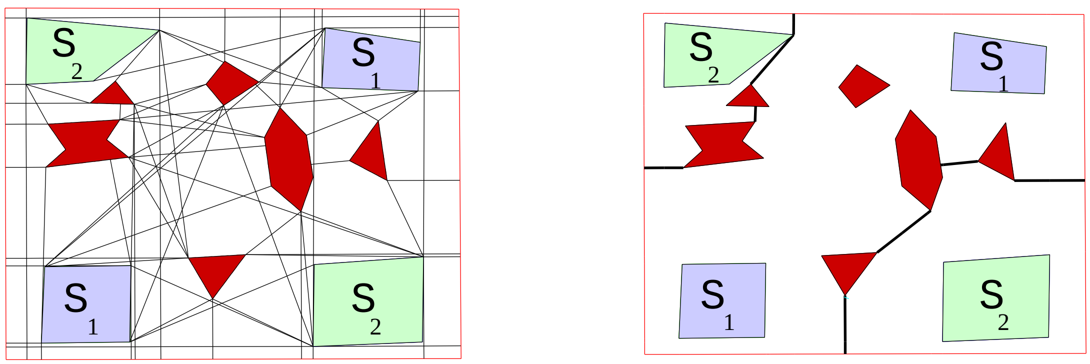
This method scales well to more complex domains. Figure 2.14 shows the algo-



(a) Barrier candidate graph.

(b) Minimum barrier

Figure 2.14: Another example for minimum barrier



(a) Barrier candidate graph

(b) Minimum barrier

Figure 2.15: Environment from Figure 2.14 less one obstacle

rithm applied to a domain with more obstacles and multiple components for $\mathcal{S}_1$ and $\mathcal{S}_2$. Figure 2.14(a) shows the barrier candidate graph, and Figure 2.14(b) shows the resulting minimum barrier. Figure 2.15 shows the effects of removing one obstacle from this sample problem. A small change to $\mathcal{W}$ can create a fundamental change in the minimum barrier. This suggests that the minimum barrier problem cannot be solved with incremental methods.

If n is the number of obstacle edges plus $\mathcal{S}_i$ edges, then there are $O\left(n^2\right)$ barrier candidates. Therefore $\mathcal{N}$ has $O\left(n^4\right)$ vertices and $O\left(n^4\right)$ edges. Since $\mathcal{N}$ is planar except for edges from v_{s_1} and v_{s_2}, of which there are $O(n)$, according to [33] the running time for finding the minimum barrier is $O(n^4 \log n)$ in the worst case and $O(n^4)$ in the average case.

2.4 Accommodating parallel edges

In Section 2.3 we said that if two obstacle edges are parallel, we include one minimal segment connecting the edges, and ignore all others. In this section, we show why doing this does not prevent the given algorithm from finding a minimum barrier.

There are an infinite number of admissible minimal segments between two parallel edges, all with the same length. These are all the segments perpendicular to both obstacle edges. For a given pair of parallel obstacle edges e_1 and e_2, let Q_M be the set of all guard configurations whose visibility regions are such minimal admissible segments, and select an arbitrary $q^* \in Q_M$. In this section we will show that for any $q \in Q_M$, and any complete barrier containing q, q can be replaced by either q^* or a set of barrier candidates outside of Q_M without increasing the total barrier length.

Define $\mathcal{M} = \bigcup_{q \in Q_M} V(q)$. Now consider the components of $\mathcal{M}$. These are the maximal rectangles containing only points in minimal admissible segments. For any two minimal segments in the same component, one can replace the other in a complete barrier while preserving the barrier. Each component is bounded by a minimal segment that touches either (1) a vertex of e_1 or e_2, (2) a vertex of $\mathcal{S}_i$, or (3) an obstacle vertex that is not in e_1 or e_2. Figure 2.16 shows an example pair of parallel edges, and the corresponding components and boundary segment types.

If the component is bounded on both sides by segments of type 1, then $\mathcal{M}$ has one component, and all segments can be replaced by q^*. Therefore it suffices to include only q^* in the barrier candidate graph. If the component is bounded on a side by a segment of type 2, this segment goes through a vertex v' of $\mathcal{S}_i$. This segment is

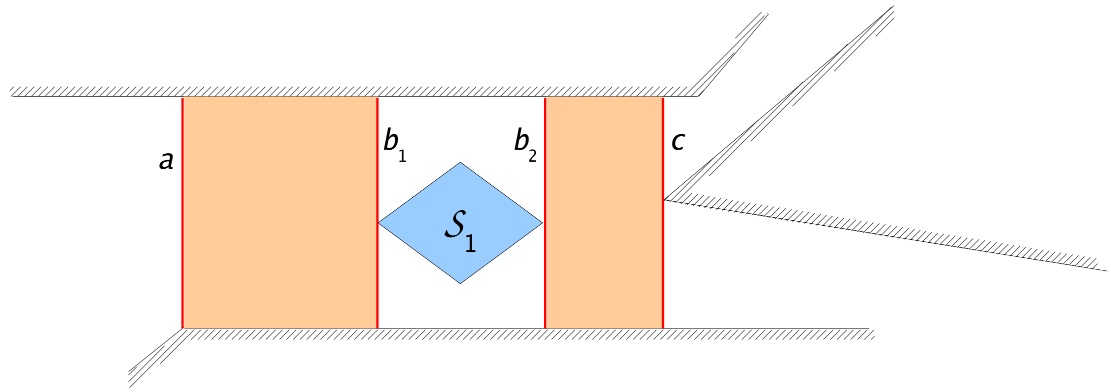


Figure 2.16: Example parallel obstacle edges. Components of $\mathcal{M}$ are shaded. Segment a is type 1, segments b_1 and b_2 are type 2, and segment c is type 3.

the union of two segments: the minimum segments from v' to e_1 and e_2, that are both tangent to $\mathcal{S}_i$ at v. Both of these segments are already in the barrier candidate graph. Therefore all minimal segments in this component can be replaced with these pre-existing segments. Similarly, if the component is bounded by a segment of type 3, it can be split into two obstacle-to-obstacle segments, at least one of which can be shortened further using the methods in the proof to Theorem 1. Therefore all of the segments in this component can be replaced by barrier candidates while decreasing the barrier length.

Therefore, when constructing the barrier candidate graph and connectivity network, it suffices to select one admissible minimal segment for each pair of parallel obstacle edges for inclusion in the barrier candidate graph. Without loss of generality, this can be the segment with the smallest x coordinate, or if the edges are vertical, the segment with the smallest y coordinate.

In the case of parallel obstacle edges, while the above theorem still holds inasmuch as every minimum barrier consists of minimal segments or tangents, it is no longer true that every minimum barrier consists of segments from the barrier candidate graph. However, since we have shown that every minimal segment can be replaced with one in the barrier candidate graph while maintaining the barrier, every minimum barrier is either composed of barrier candidates, or generated by replacing barrier candidates with the appropriate segments in Q_M. This means the given method is applicable to environments with parallel obstacle edges.

2.5 Summary

In this chapter, we have shown an efficient method for finding the minimum complete barrier for variable-length segment guards. The key to this method is the concept of barrier candidates, which show how the minimum barrier problem is reduced to the problem of searching through a small set of line segments that are easily found. The principle of barrier candidates appears throughout the later chapters of this work. Barrier candidates are key in Chapter 5, where they are used directly in constructing partial barriers.

Chapter 3

Fixed-length segment guards

The method from Chapter 2 constructs a minimum complete barrier for segment guards of variable lengths. For an integral number of guards with preset lengths, the solution is different, as is the method to find it. In [13] it is proven that for large enough guard ranges and small enough $\mathcal{S}_1$ and $\mathcal{S}_2$ components, this problem is NP-Complete. Therefore there is no known efficient algorithm for the general problem.

In this chapter we describe methods for finding the minimum complete fixed-length segment barrier. This is the deployment of the fewest guards, all of which have the same range r, that prevents all intrusions. In Section 3.1 we give an inefficient but exact method for finding the minimum complete barrier using Tarski sentences, and in Section 3.2 we give efficient methods that give complete barriers that are not minimum, and establish bounds on the costs of these barriers.

3.1 Exact solutions

The problem of definitely determining the minimum deployment with fixed-length guards is NP-Complete. This follows from [13]: for very small $\mathcal{S}_i$ components and very long guard ranges, this subproblem is NP-Complete. Therefore we do not seek algorithms that are both exact and efficient.

In this section we give a doubly-exponential solution using Tarski sentences [48]. These are first-order logic statements constructed from equalities and inequalities of polynomials over the reals with rational coefficients. We write a set of formulae that are true iff there is a set of n guards that separate $\mathcal{S}_1$ from $\mathcal{S}_2$. To find a minimum deployment, we can run this for $n = 1, 2$, etc. until we find an n such that there is a barrier for n guards but not $n - 1$ guards. We can save time by starting not at 1, but at the length of the minimum variable-length barrier divided by the length of one guard.

In Section 3.1.1, we define the terms we will be using. In Section 3.1.2, we convert elements of barrier coverage into Tarski formulae. In Section 3.1.3 we combine these elements to construct a single Tarski sentence that can be solved to answer the barrier coverage query. Section 3.1.4 gives the time complexity of this method.

3.1.1 Definitions

Here we give definitions for terms we will use throughout this section. These definitions are from Chapter 33 of [18].

- A *term* is a combination of finitely many variables and constants using the binary arithmetic operators +, -, and $\cdot$. For example, x, $x_1^2 - 2x_1x_2$, etc.
- An *atomic formula* is a combination of two terms using a binary relational operator $(=, \neq, >, <, \leq, \geq)$. For example, $x = y$, $x^2 + y^2 > 1$, etc.
- A *quantifier* is existential, $\exists$, or universal, $\forall$.
- A *Tarski formula* is a combination of atomic formulae using boolean operators $(\neg, \vee, \wedge)$ and quantifiers. If $\Phi(x, \ldots)$ is a Tarski formula, so are $\forall x\, \Phi(x, \ldots)$ and $\exists x\, \Phi(x, \ldots)$.
- A variable in a Tarski formula is *free* if it is not bound by any quantifier.
- A *Tarski sentence* is a Tarski formula with no free variables.

- Every Tarski sentence has a *Prenex form*:

$$\mathcal{Q}_1 \mathbf{x}^{[1]} \cdots \mathcal{Q}_n \mathbf{x}^{[n]} \ \phi\left(\mathbf{x}^{[1]}, \ldots, \mathbf{x}^{[n]}\right),$$

  where $\mathbf{x}^{[i]} = \left(x_1^{[i]}, \ldots, x_{n_i}^{[i]}\right)$, ϕ is a quantifier-free Tarski formula, and the $\mathcal{Q}_i$s are quantifiers, alternating between $\exists$ and $\forall$.

3.1.2 Formulae for barrier coverage

In this section we define formulae to describe pieces of the barrier coverage sentence. We will define formulae that describe intersection, point inclusion, and intrusion paths.

Intersection We define $ISECT^C(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)$ to be the Tarski formula that is true iff the line segment from (x_1, y_1) to (x_2, y_2) intersects the line segment from (x_3, y_3) to (x_4, y_4). We can write this as

$$\begin{aligned} ISECT^C(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4) = \quad & \exists t_1 \exists t_2 \, (t_1 \geq 0) \wedge (t_1 \leq 1) \\ & \wedge (t_2 \geq 0) \wedge (t_2 \leq 1) \\ & \wedge ((1-t_1)x_1 + t_1 x_2 = (1-t_2)x_3 + t_2 x_4) \\ & \wedge ((1-t_1)y_1 + t_1 y_2 = (1-t_2)y_3 + t_2 y_4) \,. \end{aligned} \tag{3.1}$$

We similarly define $ISECT^O$:

$$\begin{aligned} ISECT^O(x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4) = \quad & \exists t_1 \exists t_2 \, (t_1 > 0) \wedge (t_1 < 1) \\ & \wedge (t_2 \geq 0) \wedge (t_2 \leq 1) \\ & \wedge ((1-t_1)x_1 + t_1 x_2 = (1-t_2)x_3 + t_2 x_4) \\ & \wedge ((1-t_1)y_1 + t_1 y_2 = (1-t_2)y_3 + t_2 y_4) \end{aligned} \tag{3.2}$$

This is like $ISECT^C$, but is false if the intersection point is (x_1, y_1) or (x_2, y_2). The superscripts represent (O)pen and (C)losed.

Inclusion We define $INSIDE^P(x^*, y^*, x_1, y_1, \ldots, x_n, y_n)$ to be true iff the point (x^*, y^*) is inside the convex polygon with vertices $(x_1, y_1, \ldots, x_n, y_n)$ in counterclockwise order. If we consider the convex polygon to be the intersection of n half-planes, (x^*, y^*) is inside the polygon iff it is contained in each half-plane:

$$\begin{aligned} INSIDE^P(x^*, y^*, x_1, y_1, \quad \ldots, \quad & x_n, y_n) = \\ & \bigwedge_{j=0}^{n-1} (y^* - y_j)(x_{j+1} - x_j) \geq (x^* - x_j)(y_{j+1} - y_j), \end{aligned} \tag{3.3}$$

where we adopt the convention that $x_0 = x_n$ and $y_0 = y_n$.

Guard descriptions We define a segment guard by its endpoints $\left(x_{iA}^G, y_{iA}^G\right)$ and $\left(x_{iB}^G, y_{iB}^G\right)$. A guard is valid iff it has a length at most r, and does not intersect any obstacles except at the guard's endpoints. Let m be the number of obstacle edges, and encode each edge by its endpoints (x_{jA}, y_{jA}) and (x_{jB}, y_{jB}). We define:

$$\begin{aligned} VALID^G\left(x_A^G, y_A^G, x_B^G, y_B^G\right) &= \left(\left(x_A^G - x_B^G\right)^2 + \left(y_A^G - y_B^G\right)^2 \leq r^2\right) \\ & \wedge \bigwedge_{j=1}^{m} \neg ISECT^O\left(x_A^G, y_A^G, x_B^G, y_B^G, x_{jA}^O, y_{jA}^O, x_{jB}^O, y_{jB}^O\right) \end{aligned} \tag{3.4}$$

Notice the use of $ISECT^O$, which permits the guard's endpoints to touch the obstacles.

Start and Stop points We now build a formula to test whether a point is inside $\mathcal{S}_1$ or $\mathcal{S}_2$.

We describe $\mathcal{S}_1$ as a set of M (not necessarily disjoint) convex polygons $\left\{\left(x_{1,1}^{S_1}, y_{1,1}^{S_1}, \ldots, x_{n_1,1}^{S_1}, y_{n_1,1}^{S_1}\right), \ldots, \left(x_{1,M}^{S_1}, y_{1,M}^{S_1}, \ldots, x_{n_M,M}^{S_1}, y_{n_M,M}^{S_1}\right)\right\}$, where each polygon has vertices specified in counterclockwise order (see $INSIDE^P$ above).

Define $INS_1(x, y)$ to be true iff $(x, y) \in \mathcal{S}_1$:

$$INS_1(x, y) = \bigvee_{j=1}^{M} INSIDE^P \left(x, y, x_{1,j}^{S_1}, y_{1,j}^{S_1}, \ldots, x_{n_j,j}^{S_1}, y_{n_j,j}^{S_1} \right). \tag{3.5}$$

INS_2 is defined analogously.

Undetected paths Consider a polygonal path $(x_1, y_1, \ldots, x_P, y_P)$, where an intruder moves from (x_1, y_1) to (x_2, y_2) to (x_3, y_3) etc. until (x_P, y_P). We say a path is *valid* if it is both feasible and undetected, i.e. if it is possible to traverse this path without crossing obstacles or guards. We construct a formula to determine whether a polygonal path is valid.

Let n be the number of segment guards. We define the $VALID^P$ formula to be:

$$\begin{aligned} VALID^P(x_1, y_1, \quad \ldots, \quad & x_P, y_P) = \\ & \bigwedge_{j=1}^{m} \bigwedge_{k=1}^{P-1} \neg ISECT^C \left(x_{jA}^O, y_{jA}^O, x_{jB}^O, y_{jB}^O, x_k, y_k, x_{k+1}, y_{k+1} \right) \\ & \wedge \bigwedge_{j=1}^{n} \bigwedge_{k=1}^{P-1} \neg ISECT^C \left(x_{jA}^G, y_{jA}^G, x_{jB}^G, y_{jB}^G, x_k, y_k, x_{k+1}, y_{k+1} \right) \end{aligned} \tag{3.6}$$

$VALID^P$ is true iff $(x_1, y_1, \ldots, x_P, y_P)$ avoids all obstacles and all guards.

3.1.3 Barrier coverage Tarski sentence

The goal of the barrier coverage problem is to find a set of valid guards such that there are no valid paths that start in $\mathcal{S}_1$ and end in $\mathcal{S}_2$. We first show that, if there are m obstacle edges and n guards, it suffices to find a set of valid paths such that there are no valid polygonal paths $(x_1, y_1, \ldots, x_P, y_P)$ for $P \leq m + 2n + 2$. We then construct a Tarski sentence that is true iff there is a deployment of n guards that detects all such polygonal paths.

Lemma 1: For any environment with m obstacle edges and n deployed guards, if there is a valid path from $v_S = (x_S, y_S)$ to $v_E = (x_E, y_E)$, then there is a valid path $(x_1, y_1, \ldots, x_P, y_P)$ with $P \leq m + 2n + 2$.

Proof: For a given $\mathcal{W}$ and guard deployment, construct visibility graph [43] $\mathcal{G}^V = (V, E)$. V contains all obstacle vertices, guard endpoints, and v_S and v_E. For all $v_i, v_j \in V$, $(v_i, v_j) \in E$ iff the line segment connecting v_i to v_j crosses neither obstacle nor guard.

According to [43], every valid path through $\mathcal{W}$ is homotopic to a path through $\mathcal{G}^V$. This path through the visibility graph travels through obstacle and guard vertices; it can be perturbed slightly to become a valid path that does not touch guards or obstacles.

Consider a valid path from v_S to v_E through $\mathcal{W}$. If it contains cycles, the cycles can be removed, producing a shorter path that still connects v_S to v_E. This cycle-free path is homotopic to a cycle-free polygonal path $(x_1, y_1), \ldots, (x_P, y_P)$ through the visibility graph, where $(x_1, y_1) = (x_S, y_S)$ and $(x_P, y_P) = (x_E, y_E)$. This path can be perturbed slightly to produce a valid polygonal path of length P.

Since $(x_1, y_1, \ldots, x_P, y_P)$ is cycle-free, we can set an upper bound on P. Each (x_i, y_i) must be distinct; otherwise the path would contain a cycle. Therefore there can be at most one vertex for each obstacle or guard vertex. Since there are m obstacle vertices and n guards (each with 2 vertices), there are at most $m + 2n$ of the (x_i, y_i) vertices. Adding v_S and v_E to the path gives a total of $P \leq m + 2n + 2$.

Therefore, if there is a valid path connecting two given points, there is a valid polygonal path with at most $m + 2n + 2$ vertices that connects those two points. $\square$

This upper bound allows us to use (3.6) in a Tarski sentence. To determine if there is a complete barrier of n guards, set $P = m + 2n + 2$, and write a Tarski sentence that is true iff there exists a guard deployment such that there are no valid paths of length P. Such a deployment is a complete barrier. Using the formulae defined in the previous section, we construct this Tarski sentence:

$$\begin{aligned}
BARRIER(n) &= \exists\left[x^G_{1A}, y^G_{1A}, \ldots, x^G_{nA}, y^G_{nA}, x^G_{1B}, y^G_{1B}, \ldots, x^G_{nB}, y^G_{nB}\right] \\
&\neg \quad \exists\left[x_1, y_1, \ldots, x_P, y_P\right] \\
&\bigwedge_{j=1}^{n} \quad VALID^G\left(x^G_{jA}, y^G_{jA}, x^G_{jB}, y^G_{jB}\right) \wedge INS_1\left(x_1, y_1\right) \wedge INS_2\left(x_P, y_P\right) \\
&\wedge \quad VALID^P\left(x_1, y_1, \ldots, x_P, y_P\right)
\end{aligned} \tag{3.7}$$

Solving $BARRIER(n)$ solves a decision problem, i.e. are n guards sufficient for a given environment? To determine the minimum number of guards, determine $BARRIER(n)$ for $n = 1, 2, \ldots$ until the sentence is true. Since a fixed-length barrier cannot be shorter than the minimum variable-length barrier, it must contain at least $\frac{W}{r}$ guards, where W is the length of the minimum variable-length segment barrier. Therefore, to reduce the algorithm's running time, start evaluating $BARRIER(n)$ at $n = \left\lceil \frac{W}{r} \right\rceil$ rather than $n = 1$.

3.1.4 Complexity

To determine the complexity of this method, we first determine how many sentences, variables, etc. are needed to construct the final Tarski sentence.

First, we note that a single ISECT formula, defined in (3.2) and (3.3) requires 2 variables and 6 atomic formulae. Furthermore, all atomic formulae are polynomial in degree 1 or 2.

For this instance, there are m obstacle edges and n guards. Let s_1 be the number of polygon edges in $\mathcal{S}_1$, and let s_2 be the number of polygon edges in $\mathcal{S}_2$. As we saw in Lemma 1, $P \leq m + 2n + 2$. We break the $BARRIER$ sentence from (3.7) into its main sub-formulae, and count the variables and atomic formulae. Note that there are two sources of variables. The $BARRIER$ sentence in (3.7) introduces variables to describe guard and intruder locations. The $ISECT$ formulae (3.2,3.3) introduce t_i variables, which describe intersection locations.

- $\bigwedge_{j=1}^{n} VALID^G\left(x_{jA}^G, y_{jA}^G, x_{jB}^G, y_{jB}^G\right)$

  A single $VALID^G$ formula requires one atomic formula (to check guard length) plus m $ISECT$ formulae, each of which defines two t_i variables. Therefore n $VALID^G$ formulae require $2mn$ variables and $6mn + n$ atomic formulae.

- $INS_1\left(x_1, y_1\right) \wedge INS_2\left(x_P, y_P\right)$

  INS_1 requires one atomic formula for every polygonal edge of $\mathcal{S}_1$. Therefore, it contains s_1 atomic formulae. Similarly, INS_2 contains s_2 atomic formulae. Notice no new variables are used.

- $VALID^P\left(x_1, y_1, \ldots, x_P, y_P\right)$

  $VALID^P$ contains $(m+n)(P-1)$ $ISECT$ formulae. Of these, $m(P-1)$ check for obstacle collision, and $n(P-1)$ check for guard detection. Since each $ISECT$ formula introduces two new t_i variables, this gives a total of $2(m+n)(P-1)$ variables and $6(m+n)(P-1)$ atomic formulae.

Adding all of these values, plus the $4n + 2P$ variables already used to define $BARRIER$ yields $2m(n-1)+2n+2P(m+n+1)$ variables and $6P(m+n)-5n+s_1+s_2$ atomic formulae. For $P = m+2n+2$, this yields $2m^2+4n^2+8mn+6m+10n+4$ variables and $6m^2 + 12n^2 + 18mn + 2m - 3n + s_1 + s_2$ atomic formulae.

To determine how the variables affect complexity, we convert the sentence to Prenex form. All the new variables are introduced by $ISECT$ formulae, and all of these formulae are negated. The $\neg ISECT$ formulae that define $VALID^P$ are inside a $\neg\exists$ quantifier; these new quantifiers are of the form $\exists t_i$. The others – defined by $VALID^G$ are outside the $\neg\exists$ quantifiers. These quantifiers are the form $\neg\exists t_i$. To convert to Prenex form, every $\neg\exists\hat{\mathbf{x}}\Psi(\cdot)$ must be converted to $\forall\hat{\mathbf{x}}\neg\Psi(\cdot)$. Therefore, we write $BARRIER$ as

$$\exists\left[x_{1A}^G, y_{1A}^G, \ldots, x_{nA}^G, y_{nA}^G, x_{1B}^G, y_{1B}^G, \ldots, x_{nB}^G, y_{nB}^G\right] \forall\left[x_1, y_1, \ldots, x_P, y_P, t_1, \ldots, t_\mu\right]$$
$$\exists\left[t_1', \ldots, t_\nu'\right]\left[\phi(\cdot)\right], \tag{3.8}$$

where $\phi(\cdot)$ is

$$\bigwedge_{j=1}^{n} VALID^G\left(x_{jA}^G, y_{jA}^G, x_{jB}^G, y_{jB}^G\right) \wedge$$
$$\left(\neg INS_1(x_1, y_1) \vee \neg INS_2(x_P, y_P) \vee \neg VALID^P(x_1, y_1, \ldots, x_P, y_P)\right)$$

with all of the quantifiers removed. The variables $t_1, \ldots, t_\mu$ appear in $VALID^G$, so $\mu = 2mn$. The variables $t_1', \ldots, t_\nu'$ appear in $VALID^P$, so $\nu = 2(m+n)(P-1)$.

The Tarski sentence in this form can be solved using a method given in [44], which has complexity $s^{\Pi(k_i+1)}d^{\Pi O(k_i)}$, where the sentence has s atomic formulae with a maximum degree of d, and the ith block of qualifiers contains k_i variables. We have determined that $s = 6m^2 + 12n^2 + 18mn + 2m - 3n + s_1 + s_2$, $d = 2$, and

$$\begin{aligned} k_1 &= 4n \\ k_2 &= 2m + 2mn + 4n + 4 \\ k_3 &= 2m^2 + 4m + 6mn + 2n + 4n^2 \end{aligned}$$

Therefore, the $BARRIER$ sentence can be solved with an algorithm with a complexity of $\left(m^2 + n^2 + s_1 + s_2\right)^{O\left(m^3n^2 + mn^4\right)}$.

3.2 Approximate methods

Since finding definitively the exact minimum barrier is prohibitively expensive, we consider more efficient methods for finding a reasonably small complete barrier. We present four approaches. In Section 3.2.1, we place guards across the variable-length minimum barrier. In Section 3.2.2 we improve on this by optimizing certain guard chains, and in Sections 3.2.3 and 3.2.4 we give different ways to search for barrier candidates other than those derived from the minimum variable-length barrier.

3.2.1 Naïve variable-length barrier based method

On the simplest level, we can construct a complete fixed-length barrier from the minimum variable-length barrier – generated by the algorithm in Section 2.3 – by replacing each barrier candidate with an appropriate set of fixed-length guards.

Suppose the minimum complete variable-length segment barrier consists of ℓ barrier candidates, and the length of the jth segment is w_j. Then we construct a barrier using M guards, where

$$M = \sum_{j=1}^{\ell} \left\lceil \frac{w_j}{r} \right\rceil .$$

Since $x \leq \lceil x \rceil \leq x + 1$, we derive bounds on M as follows.

$$M = \sum_{j=1}^{\ell} \left\lceil \frac{w_j}{r} \right\rceil \leq \sum_{j=1}^{\ell} \left(\frac{w_j}{r} + 1 \right) = \frac{W}{r} + \ell, \tag{3.9}$$

where

$$W = \sum_{j=1}^{\ell} w_j$$

is the length of the minimum variable-length barrier.

We can similarly establish lower bounds on M.

$$M = \sum_{j=1}^{\ell} \left\lceil \frac{w_j}{r} \right\rceil \geq \sum_{j=1}^{\ell} \frac{w_j}{r} = \frac{W}{r}. \tag{3.10}$$

Therefore, any method that produces a result of $\left\lceil \frac{W}{r} \right\rceil$ has found an exact minimum barrier.

Combining (3.9) and (3.10),

$$\frac{W}{r} \leq M \leq \frac{W}{r} + \ell.$$

This establishes an upper bound on the gap between the number of guards found by this method and the number of guards in minimum barrier. Multiplying by r shows this gap in terms of total guard length:

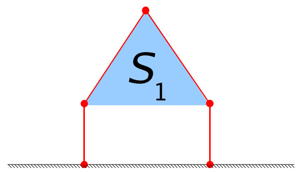
$$W \leq Mr \leq W + \ell r.$$

This gap increases linearly with r. This means that the longer the guard range, the more significant the gap between the sizes of the barrier found by this method and the minimum barrier.

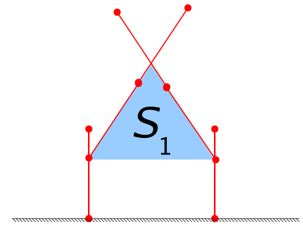
In the rest of this section, we describe different ways to potentially improve on this result. In Section 3.2.2, we optimize individual guard chains. In Sections 3.2.3 and 3.2.4 we look at different ways of choosing which barrier candidates are selected for the fixed-length guard barrier.

3.2.2 Optimizing guard chains using link distance

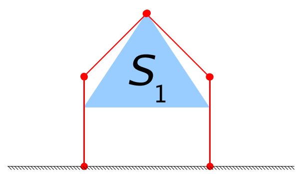
In this section, we look at improving the barrier described in Section 3.2.1 by optimizing multi-link guard chains. For an example of optimizing multi-link guard chains, see Figure 3.1. Figure 3.1(a) shows a variable-length barrier that surrounds



(a) Variable-length segment barrier



(b) Fixed-length segment barrier derived from (a)



(c) Shorter fixed-length segment barrier

Figure 3.1: Deriving fixed-length barriers from variable-length barriers

$\mathcal{S}_1$. The method of Section 3.2.1 converts this barrier into a chain of 6 fixed-length segments, as seen in Figure 3.1(b). However, an equally effective barrier can be achieved using 4 segments, as seen in Figure 3.1(c).

According to Corollary 1 in Section 2.2, variable-length guard chains that are not straight lines, like the chain in Figure 3.1(a), all involve vertices of $\mathcal{S}_1$ or $\mathcal{S}_2$. Therefore, we only optimize segments that are tangent to $\mathcal{S}_i$. Any other segments in the minimum barrier are straight lines, and cannot be locally optimized.

The problem of finding the minimum-link chain through free space connecting two features (i.e. a chain with one end on one obstacle, and the other end on the other obstacle) is related to problems involving the placement of planar multi-link robot arms. Placing the links of such an arm so that the features are connected is an inverse kinematics problem [5, 27], while ensuring guards do not cross obstacles or $\mathcal{S}_i$ is an obstacle avoidance problem [12, 46].

We view the problem as a link distance problem, and use a method modified from [47], that computes link distance between two features (points, segments, etc.).

The *link distance* between two features is the minimum number of edges in any obstacle-free polygonal path that connects them. The path that achieves this, which we call the *minimum link path*, can also be viewed as the shortest guard chain with unlimited-range guards. We will first describe the link distance method, and then show how to alter it to create minimum fixed-length guard chains.

To compute the link distance from starting feature f_s to goal feature f_g, the algorithm of [47] constructs *window partitions* to decompose $\mathcal{W}$ into regions $R_1, \ldots, R_n$ where every point in R_j is link distance j from f_s. These regions are defined based on *visibility polygons*: the visibility polygon of a feature is the set of points visible from that feature. For a single point, the visibility polygon $V_\infty(q)$, where $q = (x, y)$, is

$$V_\infty(q) = \bigcup_{r=0}^{\infty} \bigcup_{\theta=0}^{2\pi} V_{seg}(x, y, \theta, r), \tag{3.11}$$

where V_{seg} is the visibility region from Section 2.1. For a multiple-point feature f like a segment, the visibility polygon is the union of all visibility regions for every point in f.

The window partitions follow directly from the visibility polygons. Let $V_\infty(f)$ be the visibility polygon of feature f. We use V_∞ to compute the window partitions:

$$R_1 = V_\infty(f_s) \tag{3.12}$$

$$R_j = V_\infty(\partial R_{j-1}) - \bigcup_{k=1}^{j-1} R_k \tag{3.13}$$

In other words, each window partition contains all points that can be seen from the boundary of the previous partition, but are not in any earlier partitions. The algorithm stops at the first n such that R_n contains at least one point in f_g. This n is the link distance, i.e. the number of links required to connect f_s to f_g.

This algorithm can be modified to find the minimum chain of fixed-length segment guards. We seek the *fixed-range link distance*, the fewest number of edges in a polygonal path connecting f_s with f_g such that every edge in this path is of length at most r. The path that achieves this is the *minimum fixed-range path*, and is also the shortest fixed-length guard chain that connects f_s to f_g.

For fixed-length guard chains, we make three major changes to the link distance algorithm of [47]. The first change is to treat $\mathcal{S}_i$ components like obstacles. This follows from Theorem 1: any chain through $\mathcal{S}_i$ can be replaced by one with the portions inside $\mathcal{S}_i$ removed while preserving the barrier. This is illustrated in Figure 2.4. The second change is to use a different visibility function V_{circ} instead of V_∞.

$$V_{circ}\left(q\right) = \bigcup_{\theta=0}^{2\pi} V_{seg}\left(x, y, \theta, r\right), \tag{3.14}$$

Consequently, for fixed-length guard chains, the regions R_j are not polygons.

The third change is to divide each R_j into separate subwindows $R_j = R_{j,1} \cup \cdots \cup R_{j,m_j}$. Each $R_{j,k}$ is connected, and the $R_{j,k}$ regions may overlap. These subwindows reflect the fact that we are not just interested in the shortest fixed-length guard chain, but the shortest fixed-length chain homotopic to the original variable-length guard chain. If f_g appears in multiple $R_{n,k}$ subwindows, then there are multiple chains of length n from f_s to f_g that are not homotopic to each other. For each R_j we define the piecewise-connected curve γ_j to be the boundary between R_j and R_{j+1}. The curve γ_j divides into $\gamma_{j,1} \ldots, \gamma_{j,m_{j+1}}$ to match the subwindows of R_{j+1}. For each subwindow $R_{j+1,k}$, the curve $\gamma_{j,k}$ separates $R_{j+1,k}$ from R_j.

For example, Figure 3.2 gives a variable-length guard chain connecting the point f_s with the bottom horizontal edge f_g. We will use window partitions to find the minimum fixed-length guard chain that is homotopic to it. Figure 3.3 shows some of the window partitions generated in this process. Figure 3.3(a) shows that R_1 is a disk of radius r about f_s, and γ_1 is a circle. Figure 3.3(b) shows R_2 to be an annulus

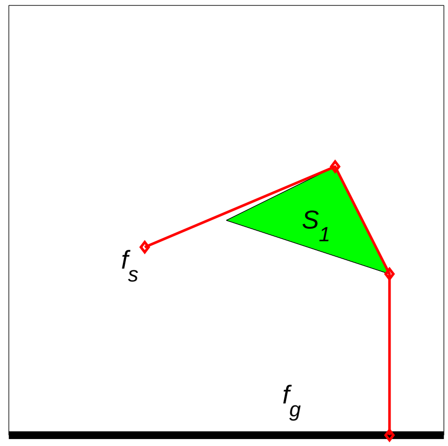


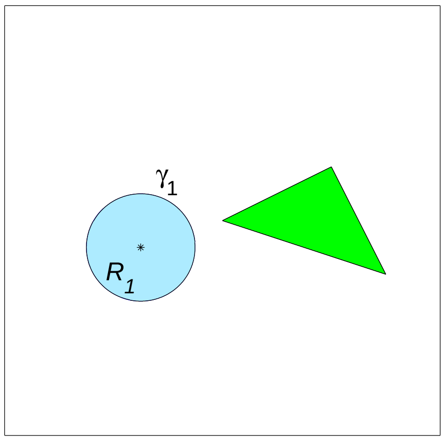
Figure 3.2: Variable-length chain

minus a section of S_1. At 3 links γ_3 divides into multiple components (Figure 3.3(c)). At 4 links R_4 divides into multiple components (Figure 3.3(d)). These components correspond to link chains that travel in opposite directions around $\mathcal{S}_1$. While $\gamma_{4,2}$ and $\gamma_{4,3}$ do intersect f_g, they represent paths that are not homotopic to the original guard chain shown in Figure 3.2. Thus the algorithm continues.

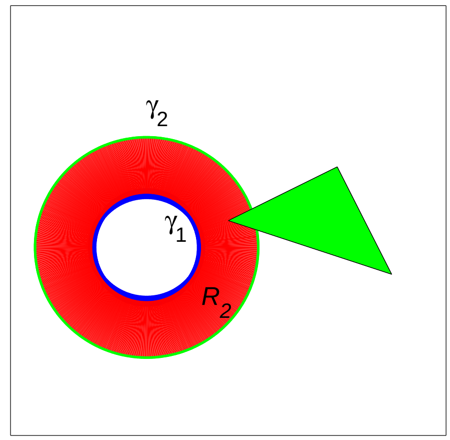
At six links, the subwindows $R_{6,1}$ and $R_{6,2}$ overlap. This corresponds to different chains to the same feature. Figure 3.4(a) shows how $R_{6,1}$ represents a clockwise guard chain, and Figure 3.4(b) shows how $R_{6,2}$ represents a counterclockwise guard chain. Both go to the same edge of $\mathcal{S}_1$.

Figure 3.5 shows the window boundaries up to 9 links, as R_9 intersects f_g and represents a chain homotopic to the given variable-length chain. Therefore, the desired fixed-length link distance for this example is 9. The desired minimum fixed-length link path is shown in Figure 3.6.

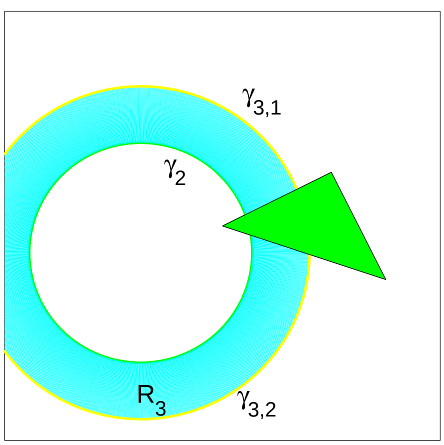
We now demonstrate how to find the window and subwindow partitions. Formally, $\gamma_j = \overline{R}_j \cap \overline{R}_{j+1}$. If γ_j is not connected, it is divided into connected components



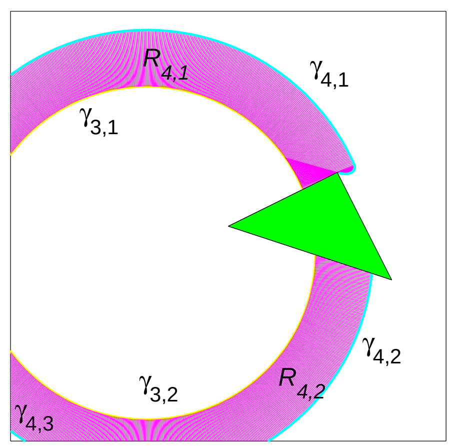
(a) One link



(b) Two links

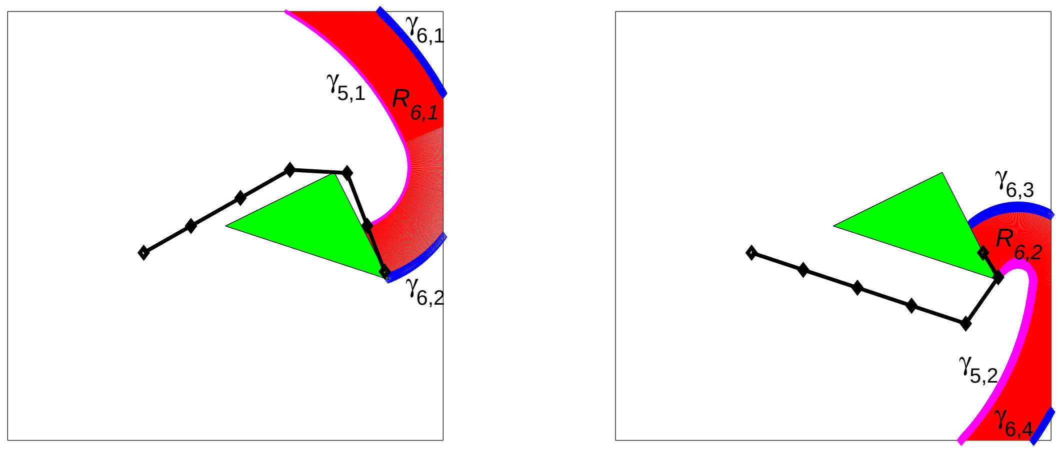


(c) Three links



(d) Four links

Figure 3.3: Window partitions for fixed-length link distance



(a) Clockwise

(b) Counterclockwise

Figure 3.4: Six links. The subwindows are not disjoint, representing two different chains.

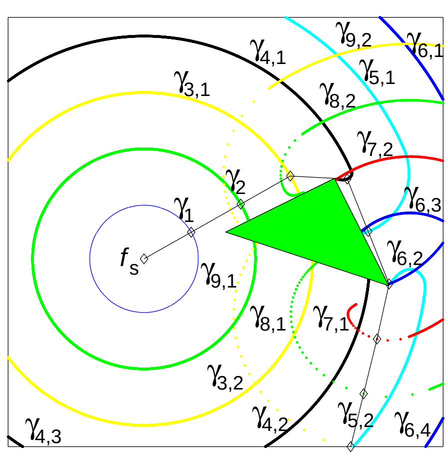


Figure 3.5: Link distance window boundaries.

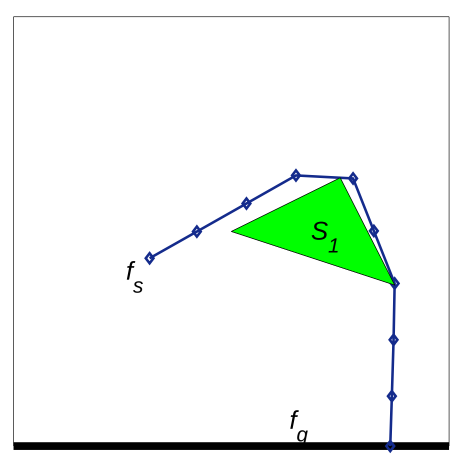


Figure 3.6: Fixed length chain

$\gamma_{j,1}, \ldots, \gamma_{j,m_{j+1}}$. Comparable to (3.13),

$$R_j = V_{circ}\left(\partial R_{j-1}\right) - \bigcup_{k=1}^{j-1} R_k \tag{3.15}$$

R_j is divided using the components of γ_{j-1} : $R_{j,k} = V_{circ}\left(\gamma_{j-1,k}\right) \cap R_j$. Here is an algorithm that, given $\gamma_{j-1,1} \ldots, \gamma_{j-1,m_j}$, determines m_{j+1}, the number of subwindows of R_{j+1}, and the m_{j+1} components $\gamma_{j,1} \ldots, \gamma_{j,m_{j+1}}$.

1. Set $m_{j+1} = 0$.

2. For each $k = 1, \ldots, m_j$:

   (a) Set $R_{j,k} = V_{circ}\left(\gamma_{j-1,k}\right) \cap R_j$.

   (b) Set $\gamma' = \partial R_{j,k} - \gamma_{j-1} - \partial \mathcal{W}$.

   (c) Let $\gamma'_1, \ldots, \gamma'_{m'}$ be the maximal connected components of γ'.

   (d) For each $u = 1, \ldots, m'$:

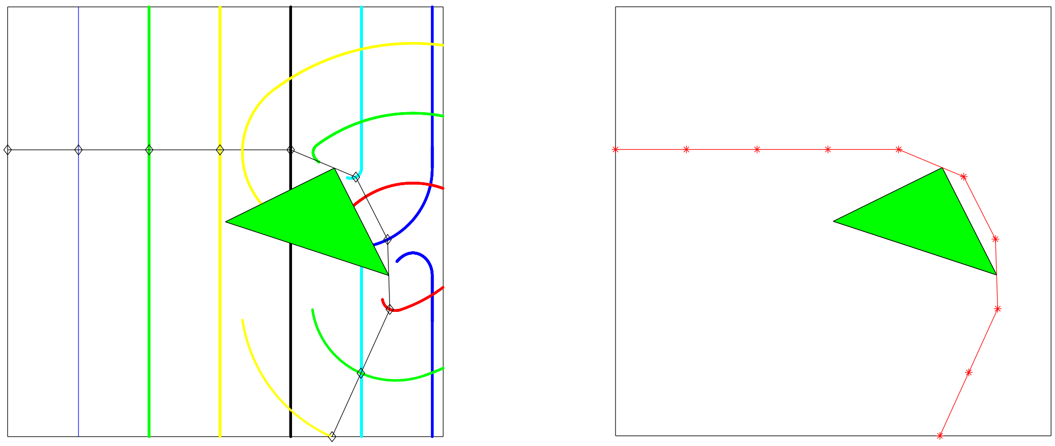
      i. Set $\gamma_{j,m_{j+1}+u} = \gamma'_u$.

   (e) Set $m_{j+1} = m_{j+1} + m'$.

Continue generating γ_j and R_j until R_j overlaps the target. Notice that if $\gamma_{j,k}$ and $\gamma_{j,k'}$ intersect, they do not recombine into one component.

Every point p in γ_{j+1} is a distance exactly r from at least one point p' in γ_j. Furthermore, the guard connecting p' to p must either (1) be normal to γ_j, or, (2) have an obstacle vertex on its interior. With this in mind, we can approximate γ_j with polygonal curves.

If the given variable-length guard chain is a loop around an $\mathcal{S}_i$ component, then $f_s = f_g$. In this case, a starting point is not defined. Therefore, select a point near a vertex, and use the above method to find the minimum chain around the component back to the starting point.



(a) Windows (b) Resulting chain

Figure 3.7: Finding a fixed-length guard chain from the left edge, around the triangle, to the bottom edge

Figure 3.7(a) gives an example where the starting feature is the far-left boundary. The resulting chain is shown in Figure 3.7(b).

Applying this method to each multi-link barrier component will often produce a shorter barrier than simply covering the variable-length barrier with fixed-length guards.

In our experiments, we used polygonal curves to approximate each γ_j. With high enough link distances resolution may be a problem; a very dense sampling of γ_1 may be necessary in order to find any appropriate points in γ_n. In this case, it may be worthwhile to view γ_j analytically.

Each $\gamma_{j,k}$ can be viewed as a continuous function $\gamma_{j,k} : [0,1] \to \mathbb{C}$, oriented counterclockwise around R_j. Then each γ_{j+1} component is composed of pieces $\eta_{j,k,u} : [0,1] \to \mathbb{C}$, where each η is derived from $\gamma_{j,k}$ via the two types of guards that connect γ_j to γ_{j+1}: normals to γ_j, and tangents to obstacles.

For the case of normals to $\gamma_{j,k}$, define

$$\eta_{j,k,0}(t) = \gamma_{j,k}(t) - ir\frac{\gamma'_{j,k}(t)}{\left|\gamma'_{i,k}(t)\right|}. \tag{3.16}$$

Here $\gamma'_{j,k}(t) = \frac{d\gamma_{j,k}}{dt}$. In other words, for each point in $\gamma_{j,k}$, move one guard length directly away from R_j. This point is in $\eta_{i,k,0}$.

Similarly, for every obstacle or $\mathcal{S}_i$ vertex $v_u \in \mathbb{C}$ such that $|\gamma_{j,k}(t) - v_u| \leq r$, we define

$$\eta_{j,k,u}(t) = \gamma_{j,k}(t) + r\frac{v_u - \gamma_{j,k}(t)}{|v_u - \gamma_{j,k}(t)|}. \tag{3.17}$$

In other words, move directly through an obstacle vertex for one guard length. This point is in $\eta_{j,k,u}$.

Given these η curves, γ_{j+1} is produced by fitting the η pieces together. This is done by removing portions of the η curves that involve moving through obstacle interiors, or appear in multiple η curves; and then fitting the remaining pieces together at their endpoints.

Figure 3.8 gives the η breakdown of γ_4 and γ_5. The $\gamma_{i,j,0}$ curves are produced by normals from $\gamma_{i-1,j}$. $\eta_{4,1,1}$ is produced by moving from $\gamma_{3,1}$ through v_1, and $\eta_{5,2,2}$ is produced by moving from $\gamma_{4,2}$ through v_2.

3.2.3 Selecting alternate barrier candidates using successive disjoint min cuts

In this section, we describe an additional potential improvement to the approximate minimum barrier. This approach also considers barrier candidates outside of the minimum variable-length barrier. We do this because it is possible that the minimum fixed-length barrier is not homotopic to the minimum variable-length barrier.

There are two major steps in this algorithm. The first is to construct a set of

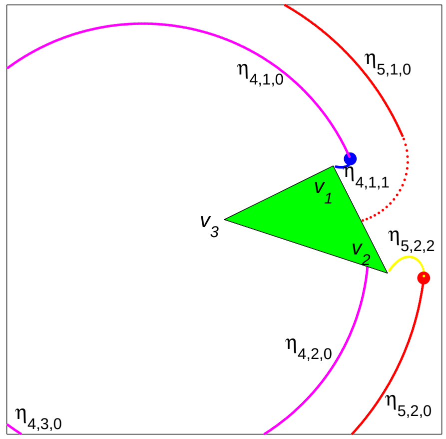


Figure 3.8: Breakdown of γ_4 and γ_5 into η curves.

complete barriers $B_0, \ldots, B_n$. The second is to incrementally combine them into a composite barrier B^* that uses the best pieces of these individual barriers.

When constructing the B_i barriers, we prefer barriers that are homotopic to the minimum barrier. These are likely to be the shortest barriers in terms of total variable-length guard length. Ideally we would construct the min cut of the connectivity network $\mathcal{N}$, followed by the second-min cut, the third-min cut etc. until the corresponding fixed-length barrier was minimized. However, the total number of edge cuts is exponential, and the problem of finding a specific-ranked cut is NP-Complete [24].

Therefore we instead seek the successive *disjoint* cuts of the connectivity network. Let E_0 be the minimum edge cut, and B_0 be the corresponding variable-length segment barrier. We define E_i to be the minimum edge cut of the connectivity network that does not include any edges from $E_0 \ldots, E_{i-1}$, and B_i to be the corresponding barrier. Each E_i can be found efficiently by taking the connectivity network, setting the weights of all edges in $E_0 \ldots, E_{i-1}$ to infinity, and finding the minimum cut of this modified network.

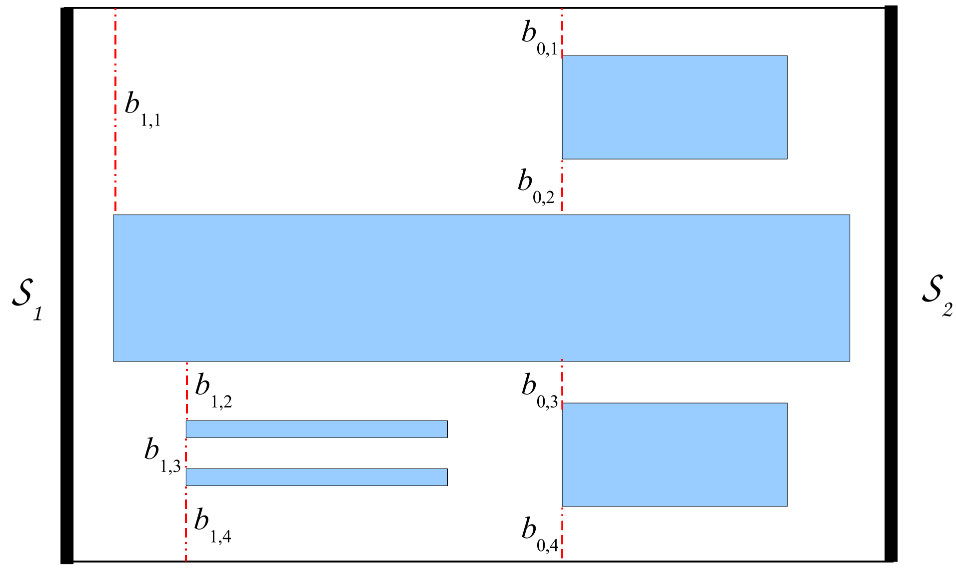


Figure 3.9: Example on combining two disjoint barriers

For any variable-length barrier B or any barrier component b, we use $\widetilde{B}$ and $\widetilde{b}$ to denote the minimum fixed-length equivalent, using the methods described above. We can use the simple method described in Section 3.2.1; we can improve these barriers by using the algorithm in Section 3.2.2. We write $\left|\widetilde{B}\right|$ to denote the number of guards in $\widetilde{B}$. Using this notation, we are looking for a B' such that B' is longer than B_0, but $\left|\widetilde{B}'\right| < \left|\widetilde{B}_0\right|$.

Given a set of disjoint barriers $B_0, \ldots, B_n$, we construct a composite barrier B^* beginning with B_0, and incrementally incorporating portions of each B_i, every time decreasing $\left|\widetilde{B}^*\right|$. For an example of this, consider Figure 3.9. B_0 is composed of all the $b_{0,j}$ segments, and B_1 is composed of all the $b_{1,j}$ segments. Suppose r is at least the length of $b_{1,1}$. Then B_1 is no improvement over B_0, as $\left|\widetilde{B}_0\right| = \left|\widetilde{B}_1\right| = 4$. Notice $\{b_{1,1}\}$ is an improvement over $\{b_{0,1}, b_{0,2}\}$, even though $\{b_{1,2}, b_{1,3}, b_{1,4}\}$ is not an improvement over $\{b_{0,3}, b_{0,4}\}$. An improved composite barrier $B^* = \{b_{0,3}, b_{0,4}, b_{1,1}\}$ contains elements from both barriers, and uses fewer guards than either.

Any B_i can be combined with the current B^* to provide a newer B^*, with a potentially smaller $\left|\widetilde{B}^*\right|$. The individual components of $B^* \cup B_i$ can be grouped into sets $C_1, \ldots, C_n$ corresponding to the connected components of $\mathcal{W} - B^* - B_i$ that contain neither $\mathcal{S}_1$ nor $\mathcal{S}_2$. If b_i and b_j are boundaries of the same component, they are in the same C_j. To accommodate intersecting guards, b_α and b_β are also in the same class if they are incident.

For each j, set $G_j = B^* \cap C_j$, and $H_j = B_i \cap C_j$. If $\left|\widetilde{G}_j\right| < \left|\widetilde{H}_j\right|$, then set $D_j = G_j$. Otherwise, $D_j = H_j$. The new B^* is $\bigcup_{j=1}^{n} D_j$. In other words, for each C_j, take the barrier component that produces a shorter fixed-length chain. Start with $B^* = B_0$, and repeat this process for $i = 1, \ldots, n$ for some n. Each new i potentially decreases $\left|\widetilde{B}^*\right|$.

For the example in Figure 3.9, the first two barriers are $B_0 = \{b_{0,1}, b_{0,2}, b_{0,3}, b_{0,4}\}$ and $B_1 = \{b_{1,1}, b_{1,2}, b_{1,3}, b_{1,4}\}$. We initialize B^* to B_0, and then combine it with B_1. $\mathcal{W} - B^* - B_1$ has two components that do not contain $\mathcal{S}_1$ and $\mathcal{S}_2$, so there are two C_j sets: $C_1 = \{b_{0,1}, b_{0,2}, b_{1,1}\}$ and $C_2 = \{b_{0,3}, b_{0,4}, b_{1,2}, b_{1,3}, b_{1,4}\}$. This reflects that any intrusion path that crosses $b_{0,1}$ or $b_{0,2}$ exactly once must also cross $b_{1,1}$ at least once; similarly any intrusion path that crosses $b_{0,3}$ or $b_{0,4}$ exactly once must cross $b_{1,2}, b_{1,3}$, or $b_{1,4}$ at least once. Since $|\{b_{1,1}\}| < |\{b_{0,1}, b_{0,2}\}|$ and $|\{b_{0,3}, b_{0,4}\}| < |\{, b_{1,2}, b_{1,3}, b_{1,4}\}|$, the new B^* is constructed from the smaller sets in each inequality. Therefore $B^* = \{b_{1,1}, b_{0,3}, b_{0,4}\}$. Notice $\widetilde{B}^*$ consists of three guards, while $\widetilde{B}_0$ and $\widetilde{B}_1$ each consist of 4.

The question remains how to determine when to stop iterating. At some point there are no further improvements, or they require too many iterations to be found. Here are some suggested terminating conditions:

- Stop after a fixed number of iterations.
- Stop when no progress is being made for two consecutive iterations.
- Stop when the length of B_i is greater than $r\left|\widetilde{B}^*\right|$, i.e. the new barrier is guaranteed not to improve the current one.

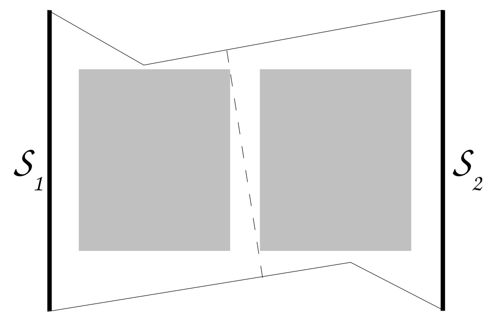


Figure 3.10: The dashed line shows a segment guard that is not in the barrier candidate graph

The results can potentially be improved further by adding certain overlooked chains to the barrier candidate graph $\mathcal{G}$ before generating $\mathcal{G}^D$ and $\mathcal{N}$. These are chains that are equivalent to shorter variable-length chains, but are actually shorter as fixed-length chains. For an example of an overlooked chain, see Figure 3.10. The dashed edge is not in the variable-length barrier candidate graph, as it does not satisfy the definition of a barrier candidate given in Section 2.2. However, for long enough guard ranges, it is shorter than its variable-length equivalent. If covering these shorter segments with fixed-length guards requires more guards than covering the one big segment, then this should be added to the graph. These candidates would be overlooked in a conventional network flows search, but may be visible once the "better" candidates are removed from the network.

Per pair of obstacle edges there may be multiple such new candidates, depending on whether there are holes between the edges. For every pair of edges, we search for a line segment that connects the two, does not cross any obstacles, and can be covered in fewer guards than the equivalent variable-length barrier component.

To search a single edge pair (e_1, e_2), first find all obstacle vertices inside the convex hull defined by the two edges. For each such vertex v, find the shortest segment through v that (a) touches both e_1 and e_2, and (b) does not cross any other obstacle edges. This is either at the minimum segment through v (if it is obstacle-free), or the line through one other obstacle vertex. Check each of these to find the

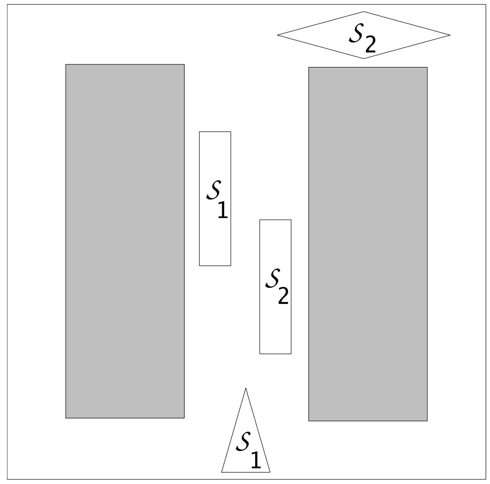


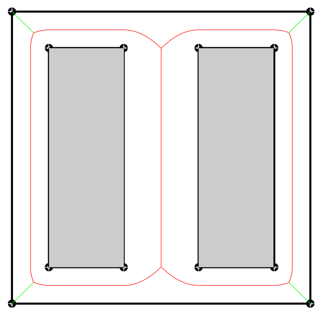
Figure 3.11: Example workspace

shortest. If it can be covered by fewer guards than the variable-length equivalent, add it to the barrier candidate graph.

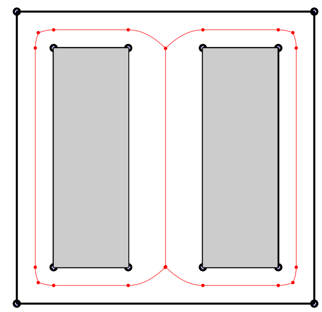
3.2.4 Set cover

We can view each barrier candidate in terms of the paths from $\mathcal{S}_1$ to $\mathcal{S}_2$ that it severs. To do this, we use the generalized Voronoi diagram [1] of $\overline{\mathcal{W}}$. Consider for example the workspace given in Fig. 3.11. If the workspace has polygonal boundaries (i.e. is a polygon with only polygonal holes), one can construct a generalized Voronoi diagram of its vertices and edges. This produces a connected undirected graph. See Fig. 3.12(a) for an example. Not all edges in this graph are useful for our purposes though. Edges that bisect adjacent features do not reflect the structure of $\mathcal{W}$ (as they connect vertices to the boundary and nowhere else). Removing these edges produces the *reduced Voronoi diagram* of $\mathcal{W}$. See Fig. 3.12(b) for the reduced Voronoi diagram of the example.

$\mathcal{S}_1$ and $\mathcal{S}_2$ can be added to the reduced Voronoi diagram as follows. Let $\mathcal{V}$ be the reduced Voronoi diagram of $\mathcal{W}$, and define $c : \mathcal{W} \to \mathcal{V}$ so that for each $p \in \mathcal{W}$, $c\,(p)$ is the point in $\mathcal{V}$ closest to p. We will apply an additional retraction to $\mathcal{V}$, based on



(a) Basic graph



(b) Relevant edges only

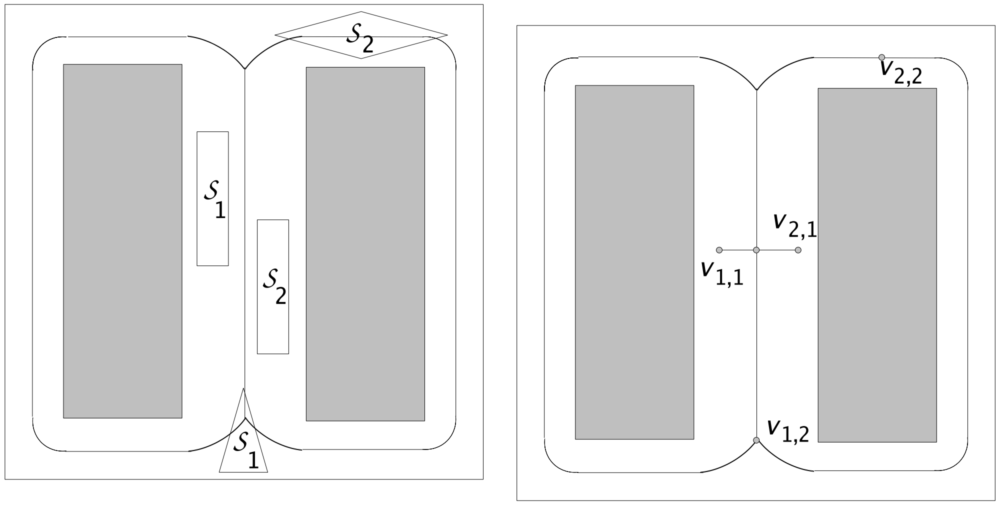
Figure 3.12: Voronoi diagram of obstacles

the components of $\mathcal{S}_1$ and $\mathcal{S}_2$. For each connected component U_a of $\mathcal{S}_1$, retract the set $c(U_a)$ to a single vertex $v_{1,a}$. Similarly, for each connected component U_b of $\mathcal{S}_2$, retract the set $c(U_b)$ to a single vertex $v_{2,b}$. If $c(U_a) \cap c(U_b) \neq \emptyset$, then they both retract to the same vertex $v^*_{a,b}$. In this case, $v_{1,a}$ and $v_{2,b}$ will be distinct degree-1 vertices connected only to $v^*_{a,b}$. See Figure 3.13.

With these vertices added, any path in $\mathcal{W}$ from $\mathcal{S}_1$ to $\mathcal{S}_2$ can be characterized as a path in $\mathcal{V}$ from a $v_{1,a}$ vertex to a $v_{2,b}$ vertex. A barrier can be viewed as a set of guards that severs every such path.

Using this modified Voronoi diagram, one can look at each barrier candidate as a weighted set. The weight is the number of guards necessary to implement it, and the set is the set of Voronoi paths from a $v_{1,a}$ vertex to a v_{2_b} vertex that the barrier candidate severs. A minimum barrier is thus the set of candidates of minimum total weight such that the union of paths is the complete set of all intrusion paths. This is a weighted version of the *set cover problem*.

The set cover problem is this: given a set $\mathcal{U}$ and a collection $\mathcal{S}$ of subsets of $\mathcal{U}$, find the smallest set of subsets whose union is $\mathcal{U}$. The weighted set cover problem gives a weight to each set, and seeks the set of subsets with minimum total weight.



(a) Before (b) After

Figure 3.13: Adding $\mathcal{S}_1$ and $\mathcal{S}_2$ to the Voronoi diagram.

If guards are unbounded, all weights are 1.

Set cover is NP-Complete [24], but there are approximate solutions. A greedy algorithm can be implemented in linear time, and gives a solution within $M(\log|\mathcal{U}|+1)$, where M is the minimum barrier size [22], and $\mathcal{U}$ is the set of homotopy classes of cycle-free intrusion paths. This is true of both weighted and unweighted [10] subsets. The current best performance of an polynomial approximation method for set cover is $(1-o(1))\log|\mathcal{U}|$ [15]. Since the number of paths is exponential in the number of internal obstacles, this approximation is linear in the number of obstacle or $\mathcal{S}_i$ edges. In some environments this is no better than the naïve approach.

Set cover can be applied alternatively by setting $\mathcal{U}$ to be the set of ordered pairs (U_a, U_b) where U_a is a component of $\mathcal{S}_1$ and U_b is a component of $\mathcal{S}_2$. For an integer k, the collection $\mathcal{S}$ of subsets of $\mathcal{U}$ is represented by cycles in $\mathcal{G}$ that contain at most k barrier candidates. The subset of $\mathcal{U}$ corresponding to a given cycle is the set of (U_a, U_b) pairs that the cycle completely separates. The weight on each subset is the number of guards in all of the barrier candidates in the cycle.

For each k there are $O\left(n^{2k}\right)$ cycles. Therefore, a greedy set cover algorithm has a running time of $O\left(kn^{4k}\log n\right)$. If there are m $\mathcal{S}_i$ components, the greedy algorithm gives a $\log m$ approximation of the smallest barrier that uses chains of at most k components.

This algorithm is polynomial for a fixed k; a good upper bound for k is ℓ, the number of barrier components in the minimum variable-length barrier. However, ℓ can be as high as $O(n)$. Therefore there are environments where this method cannot be both efficient and a good approximation. This method gives good results when k is small. For example, if $k = 1$, all guards in the minimum barrier connect points on the outer boundary without touching internal obstacles. This generally happens when guards have a long range; these are also the situations where the variable-length barrier is least efficient.

Chapter 4

Circle guards

Another useful guard in two dimensions is the circle guard. A circle guard can see in all directions up to a limited distance or to the boundary of $\mathcal{W}$. In open space, such guards have circles for visibility regions. A circle guard located at (x_j, y_j) with a range of r_j can be parameterized as $q_j = (x_j, y_j, r_j) \in \mathbb{R}^2 \times \mathbb{R}^+$. The visibility region $V_{circ}(q)$ of $q = (x, y, r)$ can be written as

$$V_{circ}(q) = \bigcup_{\theta=0}^{2\pi} V_{seg}(x, y, \theta, r), \tag{4.1}$$

where V_{seg} is the visibility region function from Section 2.1. This is the same function as in (3.14). See Figure 4.1.

Just as a segment guard with long enough range is a line guard, a circle guard with long enough range is an omnidirectional guard. Such a guard sees in all directions, up to an obstacle, with no other limitations.

In Section 4.1, we prove that the minimum circle barrier problem is intractable. In Section 4.2, we give an exact method for finding the minimum complete circle barrier by modifying the exact method for segment barriers. In Section 4.3, we discuss approximate methods. Some of these will be adapted from the segment barrier methods in Section 3.2, and others will be unique to circles.

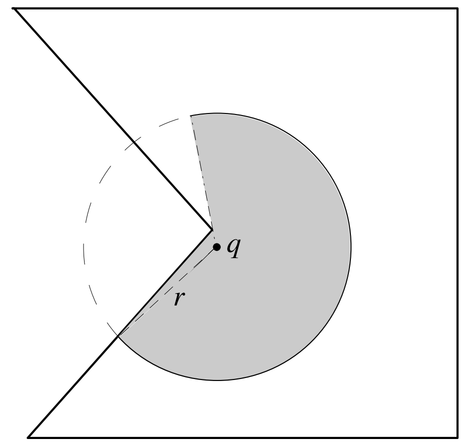


Figure 4.1: Visibility region of a circle guard at q with radius r

4.1 Intractability

In this section, we show that the minimum circle barrier problem is intractable, by showing that a subproblem is NP-Complete. This subproblem is the problem of determining if, given a workspace $\mathcal{W}$, start and stop sets $\mathcal{S}_1$ and $\mathcal{S}_2$, and positive integer M, it is possible to separate $\mathcal{S}_1$ from $\mathcal{S}_2$ with M omnidirectional guards. This decision problem, which we call **GUARD-SEP**, is equivalent to the problem of finding the size of the minimum omnidirectional barrier, which is a subproblem of minimum circle barrier.

We demonstrate that **GUARD-SEP** is NP-Complete by producing a reduction from the satisfiability problem (SAT). We show how, given a boolean expression of the form $\bigwedge_{j=1}^{p} (\ell_{j,1} \vee \ell_{j,2} \vee \ell_{j,3})$, where each $\ell_{j,k}$ is a literal of the form x_i or $\overline{x}_i$, we can create a workspace and define an M such that the boolean expression is satisfiable iff a barrier of size M exists. This reduction resembles that of [39], which converts variables into polygons that are connected with crossings and gates.

We convert each variable in the target expression into a polygon with a single hole. We then connect these polygons to gates that represent the clauses of the expression. Figure 4.2 shows the representation of a single variable, and Figure 4.3

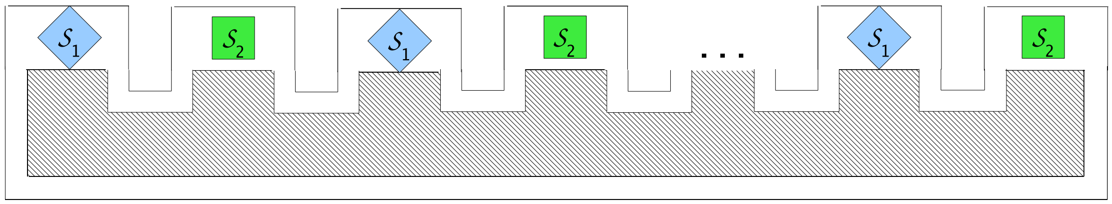


Figure 4.2: Minimum barrier representation for single variable.

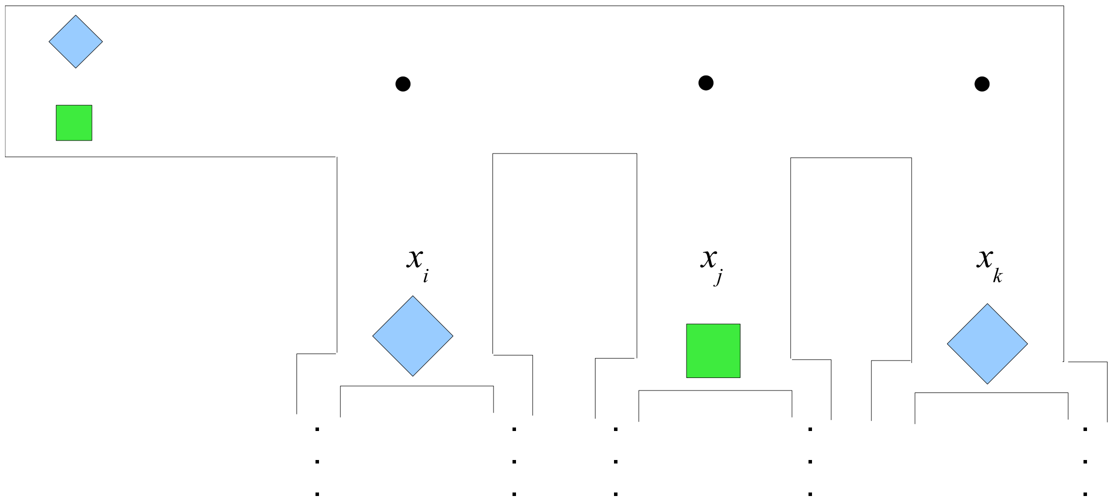


Figure 4.3: OR gate for minimum barrier. Blue diamonds are $\mathcal{S}_1$ components; green squares are $\mathcal{S}_2$ components. This gate represents the clause $(x_i \vee \overline{x}_j \vee x_k)$. To ensure minimum barrier, place a guard on one of the marked locations

shows the representation of a clause. Figure 4.4 shows a simple example of how they fit together; it illustrates the expression $(x_1 \vee \overline{x}_2 \vee x_3) \wedge (\overline{x}_1 \vee x_2 \vee x_3)$.

We now explain how the different pieces represent the variables and clauses. Each variable x_i is represented as a polygon with a single hole, as illustrated in Figure 4.2. If there are m_i components of $\mathcal{S}_1$ and m_i of $\mathcal{S}_2$, then the minimum barrier for this environment requires m_i guards, achieved by either covering every $\mathcal{S}_1$ component, or by covering every $\mathcal{S}_2$ component. We represent the values of x_i by the locations of the guards. A barrier that covers every $\mathcal{S}_1$ component represents the statement "x_i is true"; A barrier that covers every $\mathcal{S}_2$ component represents the statement "x_i is false". Both of these covers are minimum, requiring m_i guards. Any other barrier will require more than m_i guards. A system of n variables $x_1, \ldots, x_n$ where

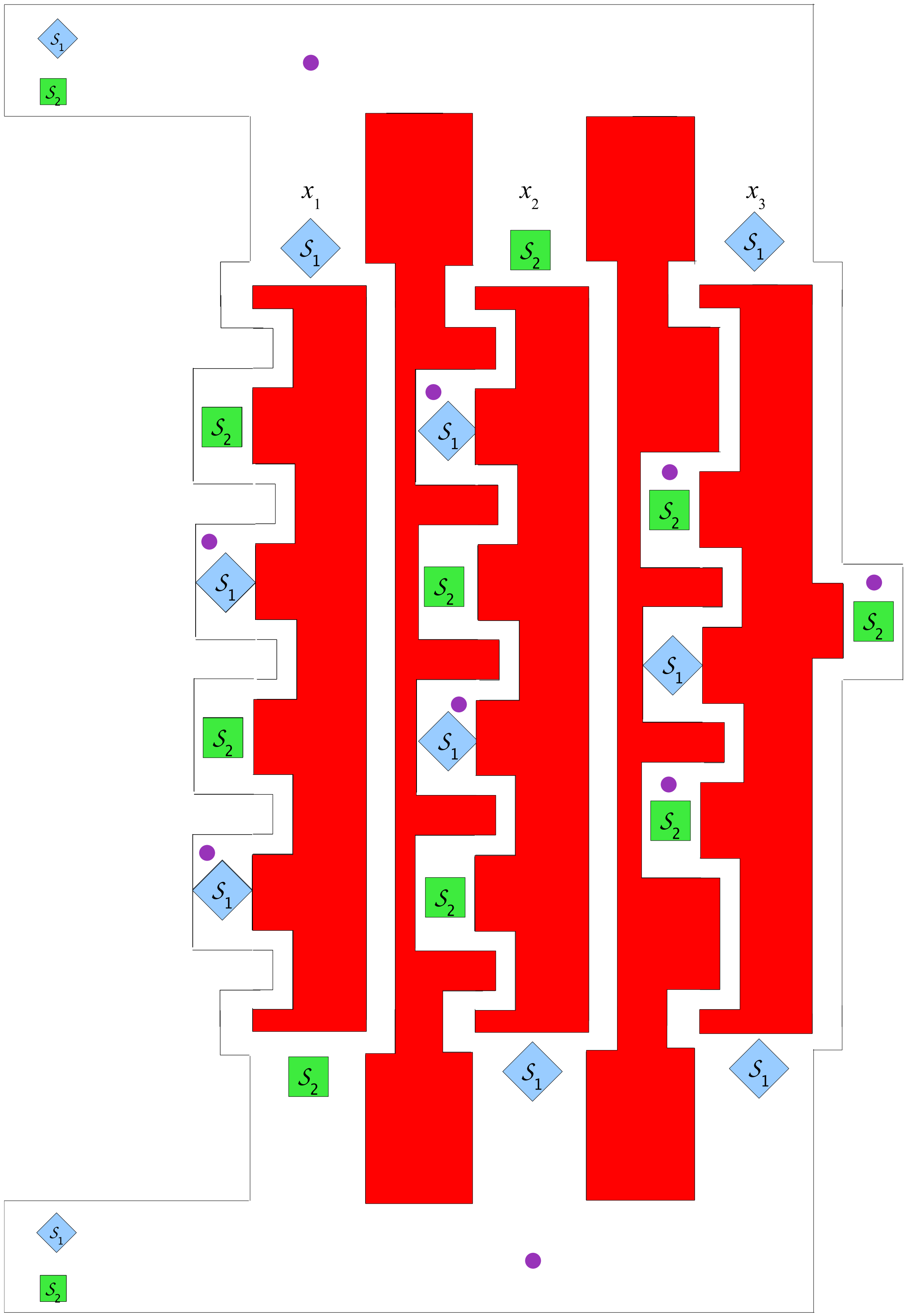


Figure 4.4: Example construction for converting a SAT problem into **GUARD-SEP**

the polygon representing x_i has m_i components of $\mathcal{S}_1$ and m_i components of $\mathcal{S}_2$ is satisfiable iff there is a complete barrier with $M = \sum_{i=1}^{n} m_i$ guards. Note that each m_i is determined by the needs of the construction. It is a function of the number of clauses the variable appears in, and the total number of variables.

To represent clauses of the form $(x_i \vee \overline{x}_j \vee x_k)$, add the gate shown in Figure 4.3, and connect it to the appropriate variables. Placing a guard at one of the three marked points separates the $\mathcal{S}_i$ components at the far left, while covering the appropriate $\mathcal{S}_i$ component of one of the variables. If at least one of the literals in the gate is true, then a barrier can be constructed without using additional guards. If, in Figure 4.3, x_i, $\overline{x}_j$, and x_k are all false, then an extra guard must be used, so the minimum barrier will be greater than M. For the barrier to be complete, every such gate must be covered, so $\bigwedge_{j=1}^{p} (\ell_{j,1} \vee \ell_{j,2} \vee \ell_{j,3})$ must be satisfiable. If the expression is not satisfiable, then there will be at least one gate that will require an extra guard, and the minimum barrier will involve more than M guards.

In the example in Figure 4.4, each variable is represented by a loop of three $\mathcal{S}_1$ components and three $\mathcal{S}_2$ components. Therefore $M = 9$. There are two gates representing two clauses; the gate at the top represents $(x_1 \vee \overline{x}_2 \vee x_3)$, and the gate at the bottom represents $(\overline{x}_1 \vee x_2 \vee x_3)$. The purple circles represent one minimum guard deployment that is a complete barrier. Since there are 9 guards, it demonstrates that the expression is satisfiable, with x_1 and x_2 as true and x_3 as false.

To ensure the complete figure exists on a single plane without losing appropriate connectivity, it is necessary to construct a way for the variable polygons described in Figure 4.2 to cross. Such a way is shown in Figure 4.5. The large square diamond in the center is an obstacle. All four components in the middle need to be separated, and this requires at least 2 guards. To avoid producing a larger barrier, use these two guards to also cover the appropriate variable's component. This can be done by placing guards at two marked points with the same label in Figure 4.5. For x_i and

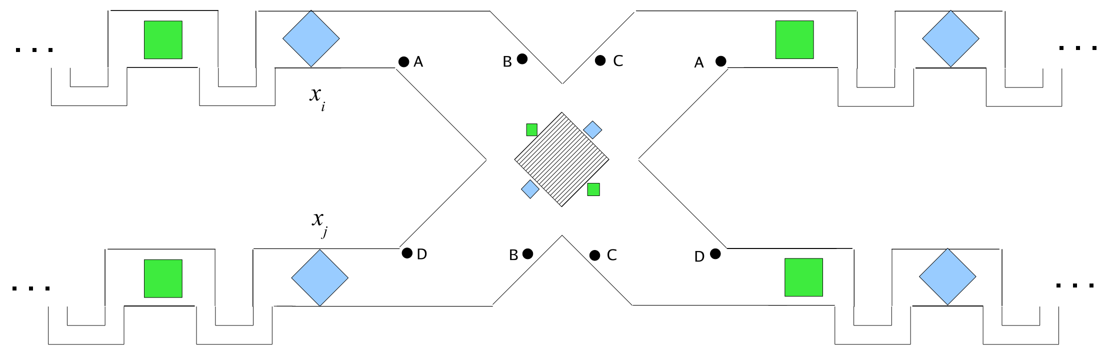


Figure 4.5: Crossing two variables. Blue diamonds are $\mathcal{S}_1$ components; green squares are $\mathcal{S}_2$ components. To ensure minimum barrier, place guards on marked locations with matching labels.

$\overline{x}_j$ are true, place guards at the locations marked "A". Similarly, place guards at "B" for x_i and x_j, "C" for $\overline{x}_i$ and $\overline{x}_j$, and "D" for $\overline{x}_i$ and x_j. This has the same effect as if the polygons did not cross, and the guards were placed to cover the appropriate components.

Therefore, we can use **GUARD-SEP** to solve 3SAT. For each x_i, the polygon that represents x_i will use m_i components of $\mathbf{S}_1$ and m_i components of $\mathcal{S}_2$. This m_i value is the number of clauses x_i appears in, plus the number of times x_i's polygon crosses another variable's polygon. To show that we can construct this instance in polynomial time, note that we need p gates, and at most $6\,(n-1)\,p$ crossings (moving a literal to the gate requires crossing at most $n-1$ other variables, at most twice per variable). Since each gate requires one pair of $\mathcal{S}_i$ components per variable, it follows that $M \leq 3p\,(2n-1)$, and the total number of $\mathcal{S}_1$ components is at most $M + 2p + 24\,(n-1)\,p \leq p\,(30n-25)$. This gives a polynomial bound for the problem size. One can show analogously that the number of obstacle edges is polynomially bounded.

Therefore, 3SAT reduces to **GUARD-SEP**, which reduces to the minimum circle barrier problem. Since 3SAT is NP-Complete, minimum complete circle barrier must also be NP-Complete.

4.2 Exact solution

In this section we give an exponential solution for the minimum deployment of circle guards using Tarski sentences. The method resembles the one in Section 3.1, and uses the same formulae from that section.

Recall that a circle guard deployment is a complete barrier if every path from $\mathcal{S}_1$ to $\mathcal{S}_2$ crosses the visibility region of some guard q_i located at (x_i, y_i). For the intrusion path to cross this guard's visibility region, it must intersect at least one point $(x_i', y_i') \in V(q_i)$. Since q sees this point, the line from (x_i, y_i) to (x_i', y_i') must be a valid segment guard.

With this in mind, a circle barrier can be viewed as a segment guard deployment where only the guard locations are initially selected. Once the intruder selects a path, the guards select directions. If there is at least one guard that has at least one direction that detects the intruder, then a circle guard at that location would detect the intruder. If this is true for every intrusion path, then the circle guard deployment would detect all intrusion paths, and must be a complete barrier.

Therefore we can take the Tarksi sentence for segment barriers, and rearrange the quantifiers to produce a sentence for circle barriers. Recall from (3.8) that the Tarski sentence for segment guards is

$$\begin{aligned}&\exists\left[x_{1A}^G, y_{1A}^G, \ldots, x_{nA}^G, y_{nA}^G, x_{1B}^G, y_{1B}^G, \ldots, x_{nB}^G, y_{nB}^G\right] \forall\left[x_1, y_1, \ldots, x_P, y_P, t_1, \ldots, t_\mu\right]\\&\exists\left[t_1', \ldots, t_\nu'\right]\left[\phi\left(\cdot\right)\right],\end{aligned}$$

where $\phi\left(\cdot\right)$ is

$$\begin{aligned}&\bigwedge_{j=1}^{n} VALID^G\left(x_{jA}^G, y_{jA}^G, x_{jB}^G, y_{jB}^G\right) \wedge\\&\left(\neg INS_1\left(x_1, y_1\right) \vee \neg INS_2\left(x_P, y_P\right) \vee \neg VALID^P\left(x_1, y_1, \ldots, x_P, y_P\right)\right)\end{aligned}$$

with all of the quantifiers removed.

Then $BARRIER^C(n)$, the analogous Tarski sentence for circle guards, can be written as

$$\exists\left[x_{1A}^G, y_{1A}^G, \ldots, x_{nA}^G, y_{nA}^G\right] \forall\left[x_1, y_1, \ldots, x_P, y_P\right]$$
$$\exists\left[x_{1B}^G, y_{1B}^G, \ldots, x_{nB}^G, y_{nB}^G, t_1', \ldots, t_\nu'\right] \forall\left[t_1, \ldots, t_\mu\right]\left[\phi(\cdot)\right].$$

The Tarski formula ϕ is unchanged. For each guard $\left(x_{iA}^G, y_{iA}^G, x_{iB}^G, y_{iB}^G\right)$, x_{iB}^G and y_{iB}^G move to an inner quantifier block, after the intrusion path is selected, while x_{iA}^G and y_{iA}^G remain in the outer block. This is because $\left(x_{iA}^G, y_{iA}^G\right)$ represents a circle guard's location, and $\left(x_{iB}^G, y_{iB}^G\right)$ represents the point where the guard detects the intruder. Similarly, the t_i variables used to describe intersection with guards have been moved inward.

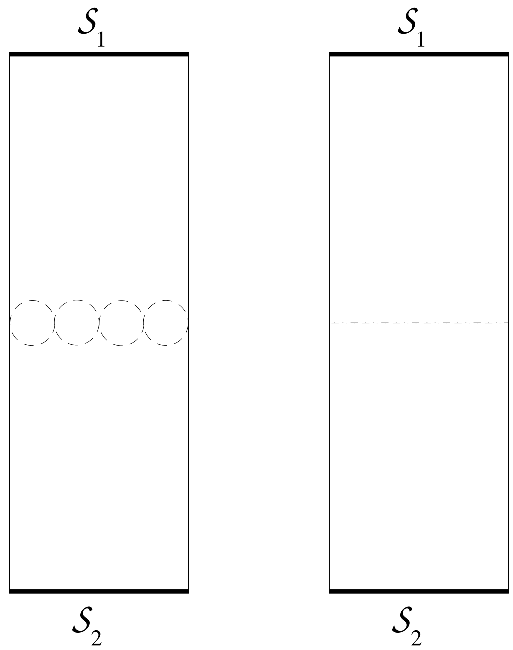
While the number of variables and atomic formulae are unchanged, the k_i values are different. The new values are

$$\begin{aligned} k_1 &= 2n \\ k_2 &= 2m + 4n + 4 \\ k_3 &= 2m^2 + 4m + 6mn + 4n + 4n^2 \\ k_4 &= 2mn \end{aligned}$$

This gives a time complexity of $(m^2 + n^2 + s_1 + s_2)^{O\left(m^4n^2 + mn^5\right)}$.

4.3 Approximate methods

In this section, we describe two different ways of efficiently finding circle barriers that approximate the minimum circle barrier. Section 4.3.1 describes methods for finite-



(a) Circles (b) Segments

Figure 4.6: Equivalent circle and segment barriers for a simple corridor

range circles; these methods are modified from segment guard methods. Section 4.3.2 defines methods for unbounded (omnidirectional) circle guards.

4.3.1 Finite-range circles: the circle-segment equivalence

Because circle guards can be seen as combinations of many segment guards, they can be used much like segment guards. One can place circles across segment barriers (or segments through circle barriers) to produce the same result (see Figure 4.6). With that in mind, we can use approximate methods for segment barriers to create circle barriers.

However, there are two cases in which circles are more powerful than segments. Once we identify these, we can adjust the segment-based algorithm by either (a) adding such circles as barrier candidates to be searched, or (b) using them when converting variable-length barriers to fixed-length barriers.

The rest of this section describes these cases, and how they can be addressed.

Polygonal curve barrier components

Consider a segment barrier component that is a chain of guards that do not lie along a straight line. For examples, see Figures 3.1 and 4.7. In some cases it is possible that placing a circle centered at a vertex of the chain will produce a smaller circle barrier than the straightforward segment-to-circle map described above. In such cases, a chain of n guards of length r can be viewed as a chain of $2n$ guards with length $r/2$. Placing circles appropriately yields n circles of radius $r/2$, i.e. diameter r. Figure 4.7 is an example of this. Eight circles are used, but the segment barrier is nine segments long.

To find such cases, every time the resulting segment barrier has a chain that does not lie along a straight line, break it into segments with half the length, and find the equivalent circle barrier. This will save at most one guard per chain component.

Circles that are more powerful

The second case to consider is one in which a circle guard does the work of multiple segment guards. A large enough circle or pair of such circles can completely cover a small enough polygonal $\mathcal{S}_i$ component, while at least three segments would be necessary. Finding such circles is simply a matter of checking each $\mathcal{S}_i$ component for size and shape.

Similarly, a large enough circle can guard an entire intersection, a task that can require multiple segment guards, no matter how high their ranges. See Figure 4.8 for an example of this. We now show that such a circle must touch at least two different obstacles in order to be more powerful than a segment.

Lemma 2: A circle that touches at most one obstacle[1] is no more powerful than

[1] When we count obstacles a circle touches, we are not counting components in $\overline{\mathcal{W}}$, but compo-

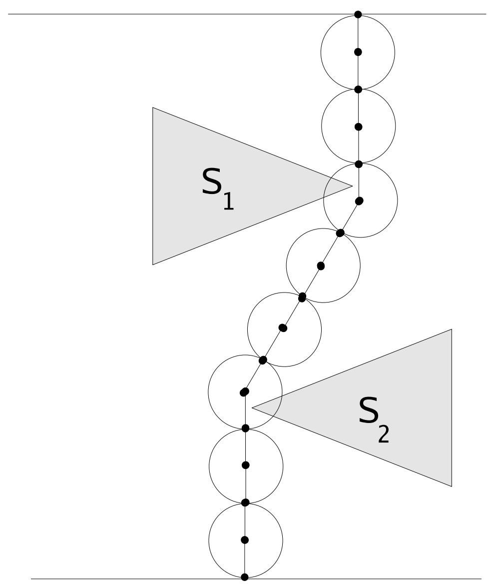


Figure 4.7: The circle and segment barriers here are equivalent.

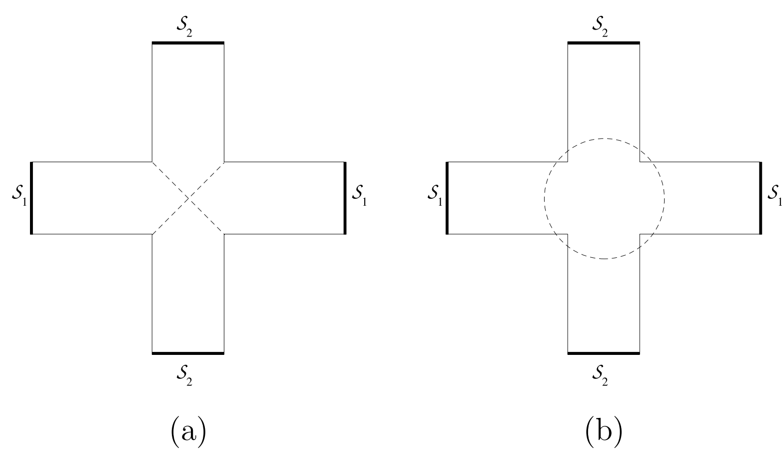


Figure 4.8: One circle does the work of two segments

a segment of the same length.

Proof: We use a proof analogous to components of Theorem 1. If a segment barrier separates three or more regions, then the barrier can be shortened by removing segments or shortening the segment chain. Figures 2.6 and 2.7 show one example of each.

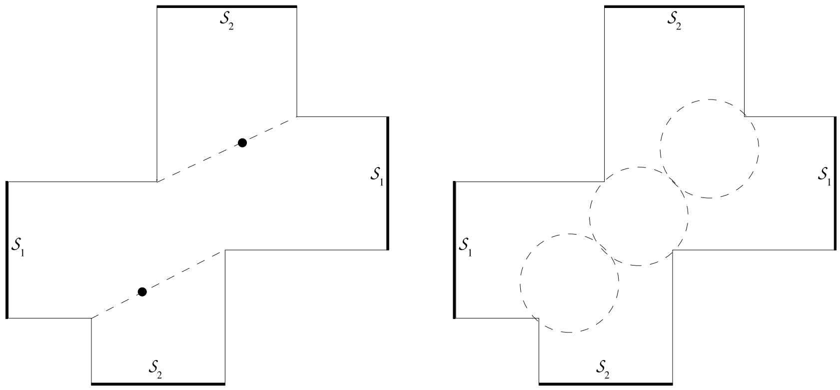
If a circle guard touches one obstacle and no other guards, it is ineffective as a guard; it can be removed without affecting the barrier.

Consider a chain of multiple circle guards connecting two obstacles such that each circle touches at most one obstacle. A circle guard at the end of such a chain can be replaced by a segment connecting the next circle to the obstacle. This can also be done for a circle guard that touches no obstacles. Consider a guard in the middle of the chain that touches an obstacle different from the ends of the chain. Such a chain separates three different regions from each other. Since the barrier only needs to separate two classes of regions (one for $\mathcal{S}_1$, one for $\mathcal{S}_2$), at least one pair does not need to be separated. Therefore, either part of the chain can be removed while preserving the barrier (compare to Figure 2.6), or the guard can be moved away from the obstacles (compare to Figure 2.7).

The above process can be repeated for every guard until all that remains is a set of segment guards. At no time does the number of guards increase. Therefore, circle guards that touch a single obstacle never offer an advantage over segment guards. □

Therefore, only circle guards that touch two or more obstacles can be more powerful than segment guards. To find these, we look for barrier candidates that are shorter than one segment. If there are no such candidates, then there are no circle guards that are more powerful than segment guards. Furthermore, such short

nents inside the circle. Using this definition, a circle that touches two non-adjacent reflex vertices of the same obstacle is touching two obstacles, not one.



(a) (b)

Figure 4.9: Three circles do the work of four segments

candidates must be near at least one other short candidate, so that they can all be covered by one circle.

Furthermore, the number of obstacles a single guard touches is relevant. Unlike the four-obstacle guard of Figure 4.8, a circle guard that touches two obstacles requires a chain of at least three circle guards to be advantageous (see Figure 4.9), while a circle guard that touches three obstacles requires a chain of two circle guards (see Figure 4.10).

4.3.2 Omnidirectional guards

Omnidirectional guards are always more powerful than line guards, as they always satisfy the condition from the previous section that they see two or more features at once. Therefore, we construct an approximate method unique to this type of guard.

The problem of omnidirectional barriers closely resembles the *Art Gallery Problem* [26]. This is the problem of placing the minimum number of omnidirectional guards to cover an entire polygon. An n-sided polygon can always be covered with at most $\lfloor \frac{n}{3} \rfloor$ guards [9]. This bound is tight. The art gallery problem is NP-Complete [38]. The minimum circle barrier problem for omnidirectional guards differs from the art gallery problem in that the guards do not need to see the entire

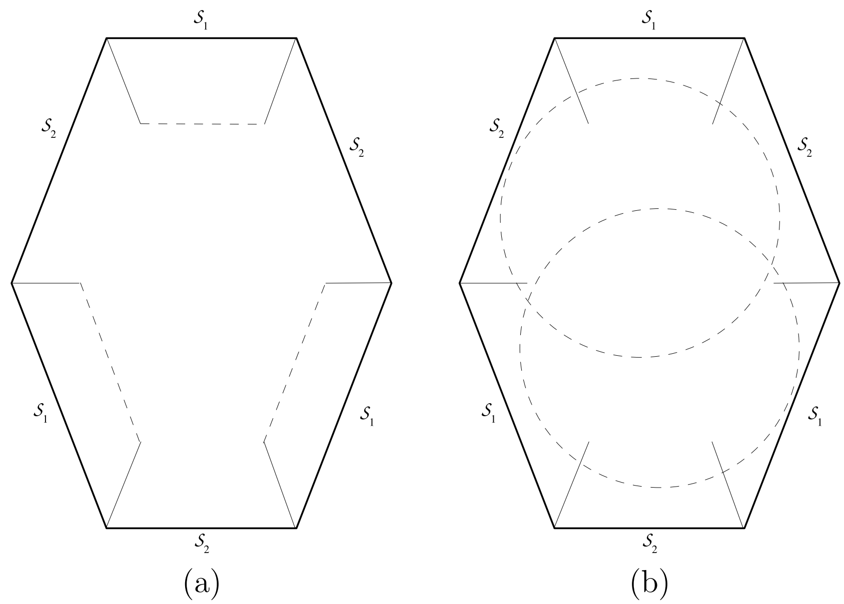
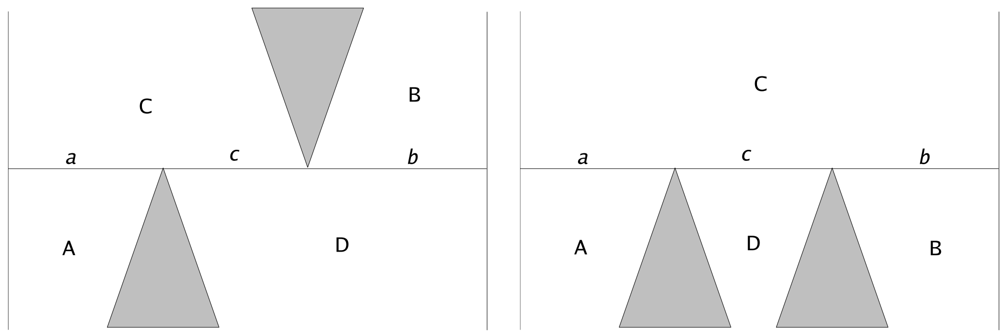


Figure 4.10: Two circles do the work of three segments

space; they just need to see enough to protect $\mathcal{S}_2$.

To determine what possible guarding power is available to omnidirectional guards in $\overline{\mathcal{W}}$, we look at edges in the reduced visibility graph [43] of $\overline{\mathcal{W}}$. These are bitangents of obstacles, and can be either separating tangents (see Figure 4.11(a)), or supporting tangents (see Figure 4.11(b)). In either case, extend each bitangent outward from each point of tangency until hitting an obstacle again. The extensions form a *bitangent complement*. See [50] for a similar use of these. The bitangent and its complement divide the workspace into regions; which barrier candidates a guard can cover depend on which region it is in. In both examples in Figure 4.11, a guard at region A cannot watch b, nor can a guard at B watch a. But guards at C and D can watch all three sections of the bitangent [2]. The guard's power changes fundamentally at the bitangent complements, and nowhere else. Furthermore, in Figure 4.11(a), if the left grey region is a component of $\mathcal{S}_1$ or $\mathcal{S}_2$ and not an obstacle, then a guard in A and C covers the same barrier candidates. Which barrier candidates

[2]A guard at D, for example, does not need to see all of a; it suffices to see the entirety of some segment that is homotopic to a.



(a) Separating tangent (b) Supporting tangent

Figure 4.11: An omnidirectional guard's power depends on what side of the bitangent it lies.

the guard covers only changes when the guard crosses b. In Figure 4.11(b), if either grey region is an $\mathcal{S}_i$ component, then the guard covers the same barrier candidates, regardless of which region it is in.

We construct an *extended visibility graph* using the algorithm for constructing a reduced visibility graph. For every edge of the reduced visibility graph, the rays extending in either direction from the edge's endpoints are added to the extended diagram. Bitangents to start or stop sets only extend on the obstacle's side, and only for separating tangents. See Figure 4.12 for an example. These new edges produce a decomposition of $\mathcal{W}$. Guards in the same region in the decomposition detect the same intrusion paths. For every region, determine which intrusion paths it guards. Then apply the set cover methods described in Section 3.2.4. This gives a polynomial time approximation that is within a multiplicative factor of $\log n$ of the minimum circle barrier.

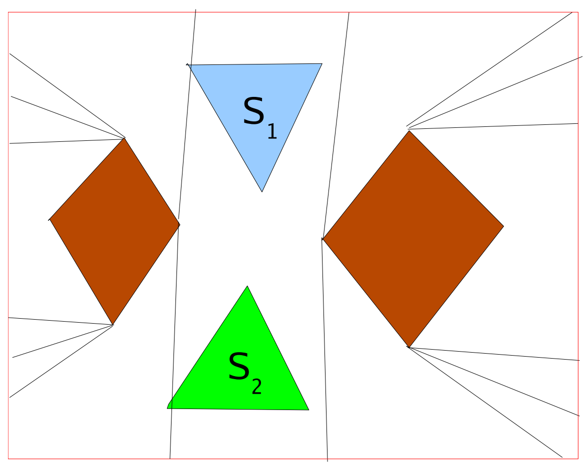


Figure 4.12: Extended visibility graph

Chapter 5

Partial coverage

In previous chapters, we have considered cases for which an adequate number of guards are available to form a complete barrier. We now turn our attention to situations for which this is not the case. When a complete barrier is not possible, we must instead determine how to best use the limited available resources and construct a *partial barrier*. To determine the best partial barrier, we need to determine how to evaluate a partial barrier. Area and sweep coverage have a clear definition of partial coverage: the area covered as a fraction of total workspace area. However, this definition does not apply to barrier coverage. Ideally partial barrier coverage would reflect measures like the percentage of intrusion paths that are blocked or the probability of detecting an intruder. In this chapter we give a precise way to evaluate the quality of a partial barrier, and give specific strategies for the case of variable-length segment guards.

We will define partial barrier coverage as the probability that an intruder attempting to reach $\mathcal{S}_2$ from $\mathcal{S}_1$ will be detected by at least one guard. For a total barrier this value is clearly 1. Besides the values that affected full barrier coverage ($\mathcal{W}$, $\mathcal{S}_1$, $\mathcal{S}_2$, and the guard deployment), we must now also consider the strategies of the guards and intruder. The intruder seeks to select paths that avoid the guards, while the guards seek to select deployments that detect the intruder. Because of this strategy component, we model the partial barrier coverage as a two-player noncooperative zero-sum game of the intruder versus the guards.

This section is laid out as follows. In Section 5.1 we will summarize the ideas from noncooperative game theory that will be used throughout this section and others. In Section 5.2, we show how to apply these concepts and methods to partial coverage in very simple environments. In Section 5.3 we generalize this to produce strategies for guards and intruders in any polygonally-bounded environment.

5.1 Game theory concepts

In this section, we review some relevant concepts of two-player zero-sum noncooperative game theory, and show how to apply these to partial barrier coverage. We use notation and ideas from [2].

We describe partial coverage as a two-player static zero-sum game between the intruder **P1** and the deployer of the guards **P2**. For **P1** an *action* is a path from $\mathcal{S}_1$ to $\mathcal{S}_2$. For **P2**, an action is a guard deployment. The *outcome* of this game is 0 if the intruder travels its path from $\mathcal{S}_1$ to $\mathcal{S}_2$ undetected, and 1 if the guards detect him. **P2**'s goal is to maximize the outcome; **P1**'s goal is to minimize it.

To analyze these games, we determine the optimal strategies for both players. We start with *pure strategies*, wherein each player acts deterministically. Since the game has only one stage, a pure strategy is equivalent to a single action that is always played. Let γ be an intrusion path from $\mathcal{S}_1$ to $\mathcal{S}_2$, and let q be a guard deployment. We define $A(\gamma, q)$ to be the outcome of the game when **P1** selects γ and **P2** selects q. To minimize this outcome in the worst case, **P1** will select a γ^* that minimizes $\max_q A(\gamma^*, q)$. This guarantees an outcome of at most

$$\overline{V}(A) = \min_\gamma \max_q A(\gamma, q)\,. \tag{5.1}$$

$\overline{V}$ is called the *upper value* of A. Similarly, to maximize this outcome in the worst case, **P2** will select a q^* that maximizes $\min_\gamma A(\gamma, q^*)$. This guarantees an outcome of at least

$$\underline{V}(A) = \max_{q} \min_{\gamma} A(\gamma, q). \tag{5.2}$$

$\underline{V}$ is called the *lower value* of A. If both players play these strategies, the game's outcome will always be between $\underline{V}$ and $\overline{V}$. If $\overline{V} = \underline{V}$, then γ^* and q^* form *equilibrium strategies* for A. The pair (γ^*, q^*) is also called a *saddle point* for *pure strategies*. The value of such a game is $V = \overline{V} = \underline{V}$. Each player can guarantee an outcome of V or better by playing his equilibrium strategy, but cannot guarantee anything better. If a player plays a strategy that is not an equilibrium, then it is possible for the outcome to be worse for that player.

Equivalently, (γ^*, q^*) is a saddle point iff for all γ and q,

$$A(\gamma^*, q) \leq A(\gamma^*, q^*) \leq A(\gamma, q^*). \tag{5.3}$$

In other words, a strategy is an equilibrium strategy iff neither player can improve his outcome by unilaterally deviating from the strategy.

For every partial coverage game, every guard deployment has an intrusion path that avoids it. Therefore, $\underline{V} = 0$. Similarly, every intrusion path has a guard deployment that detects it. Therefore, $\overline{V} = 1$. Since $\underline{V} < \overline{V}$, there are no saddle points for pure strategies.

Since pure equilibrium strategies do not exist for partial barrier games, to find equilibrium strategies it is necessary to use *mixed strategies*. A mixed strategy is defined as a probability distribution over a player's possible actions. Again following the notation of [2], we use y_i to denote the probability that **P1** plays the path γ_i, and z_j to denote the probability that **P2** plays the deployment q_j. When mixed strategies are used, these strategies can be evaluated by determining $E_{y,z}\{A\}$, the expected value of the outcome with respect to the probability distributions represented by y and z. When using mixed strategies in partial coverage problems, we define the *coverage value* to be this expected outcome. Thus, the coverage value is equal to the probability that the guards detect the intruder. As above, **P1** seeks to minimize this value, while **P2** seeks to maximize it.

The mixed strategies analogies of (5.1), (5.2), and (5.3) are

$$\overline{V}_m(A) = \min_y \max_z E_{y,z}\{A\} \tag{5.4}$$

$$\underline{V}_m(A) = \max_z \min_y E_{y,z}\{A\} \tag{5.5}$$

$$E_{y^*,z}\{A\} \le E_{y^*,z^*}\{A\} \le E_{y,z^*}\{A\}. \tag{5.6}$$

Unlike pure strategies, for mixed strategies we are guaranteed $\overline{V}_m(A) = \underline{V}_m(A) = V_m(A)$, and that equilibrium strategies y^* and z^* exist for any A.

5.1.1 Discrete notation

If the players both have a finite, discrete set of possible actions (as they will in some of the motivating examples below), we can represent the game with a matrix $A = \{a_{ij}\}$, where $a_{ij} = A(\gamma_i, q_j)$. If **P1** plays mixed strategy $y = [y_1, \ldots, y_m]^T$, and **P2** plays mixed strategy $z = [z_1, \ldots, z_n]^T$, then the expected outcome of this strategy is

$$E_{y,z}\{A\} = \sum_i \sum_j y_i a_{ij} z_j = y^T A z.$$

The equilibrium strategies are written y^* and z^*, and satisfy

$$\begin{aligned}
\overline{V}_m(A) &= \min_y \max_z y^T A z \\
\underline{V}_m(A) &= \max_z \min_y y^T A z \\
y^{*T} A z \le y^{*T} A z^* &\le y^T A z^*.
\end{aligned}$$

5.2 One corridor

In this section we show the equilibrium strategies for a single corridor with $\mathcal{S}_1$ at one end and $\mathcal{S}_2$ at the other. We will first use simpler examples to motivate the final strategies, which we will give in Section 5.2.3. The simpler examples can be represented with matrices and easily fully analyzed, but give insights into how to construct solutions for more general examples with infinite sets of possible actions.

5.2.1 Discrete, vertical corridor

To discretize the problem of partial barrier coverage in a simple, one-corridor environment, consider a row of n doorways in the middle of the corridor, such that the intruder must pass through one doorway to reach $\mathcal{S}_2$. The guards are deployed at m of those doorways; if the intruder crosses a door that a guard is watching, the intruder will be detected. Figure 5.1 depicts an example where $n = 5$ and $m = 2$. With this model, **P1**'s choices are discrete (pick a doorway), as are **P2**'s (pick which m doors to guard). Also, since the guards are only at the doorways, the paths the intruder takes to and from the doorways are irrelevant to this problem.

The example of Figure 5.1 can be represented with the matrix

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}. \tag{5.7}$$

In (5.7), each row of A corresponds to a door (to be chosen by **P1**), and each column of A corresponds to a possible placement of guards (to be chosen by **P2**). It follows that **P2** has 10 actions. For example, the first column of A corresponds to

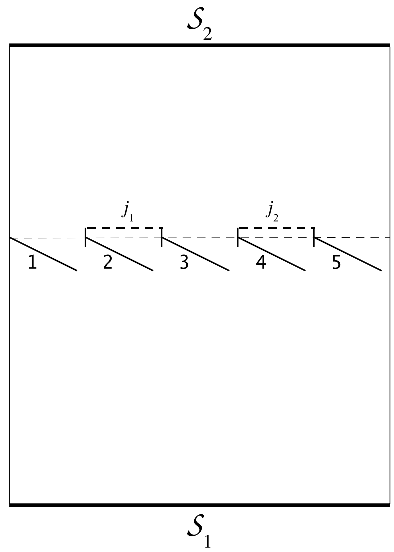


Figure 5.1: Discretized corridor with unit guards

placing guards at doors 1 and 2, the second column to placing guards at 1 and 3, etc. Using this representation of the game $a_{ij} = 1$ if **P1** chooses door i, and this door is guarded by the **P2** strategy corresponding to the jth column of A. Otherwise a_{ij}=0.

When using a mixed strategy, **P1** chooses y to minimize $\max\limits_{z} y^T Az$, in which y and z are probability distributions over actions, as described in Section 5.1.1. To understand this minimization, note that

$$y^T A = [y_1 + y_2, y_1 + y_3, y_1 + y_4, y_1 + y_5, y_2 + y_3, y_2 + y_4, y_2 + y_5, y_3 + y_4, y_3 + y_5, y_4 + y_5] \, . \tag{5.8}$$

Here $y^T Az$ contains $y_{i_1} + y_{i_2}$ for every $i_1, i_2 \in \{1, \ldots, 5\}$. Therefore the maximum component of $y^T A$ is the sum of the largest two y_i values, i.e.

$$\max_z y^T Az = \max_j \left(y^T A\right)_j = \max_i y_i + \max_{i \neq \arg\max\limits_j y_j} y_i. \tag{5.9}$$

Since $\sum\limits_{i=1}^{5} y_i = 1$, The value in (5.9) is minimized by making all y_i values equal to $\frac{1}{5}$. Therefore $\overline{V}_m(A) = \min\limits_y \max\limits_z y^T Az = \frac{2}{5}$.

By a similar process for **P2**, first note

$$Az = \begin{bmatrix} z_1 + z_2 + z_3 + z_4 \\ z_1 + z_5 + z_6 + z_7 \\ z_2 + z_5 + z_8 + z_9 \\ z_3 + z_6 + z_8 + z_{10} \\ z_4 + z_7 + z_9 + z_{10} \end{bmatrix}. \tag{5.10}$$

Every z_j appears in two elements of Az, corresponding to the two doorways covered by deployment j. For example, z_1 appears in the first and second elements of Az, since $j = 1$ covers doorways 1 and 2. Since the z_j values sum to 1, the elements of Az always sum to 2. We can maximize $\min_y Az$ by making all elements of Az equal. This happens when

$$Az = \begin{bmatrix} \frac{2}{5} \\ \frac{2}{5} \\ \frac{2}{5} \\ \frac{2}{5} \\ \frac{2}{5} \end{bmatrix}. \tag{5.11}$$

There are many choices for z that produce this. Some examples are:

$$z = \begin{bmatrix} 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \\ 1/10 \end{bmatrix}, \begin{bmatrix} 1/5 \\ 0 \\ 0 \\ 1/5 \\ 1/5 \\ 0 \\ 0 \\ 1/5 \\ 0 \\ 1/5 \end{bmatrix}, \begin{bmatrix} 0 \\ 1/5 \\ 1/5 \\ 0 \\ 0 \\ 1/5 \\ 1/5 \\ 0 \\ 1/5 \\ 0 \end{bmatrix}, \ldots \tag{5.12}$$

Any strategy that covers each doorway with a $\frac{2}{5}$ probability satisfies this. Therefore,

$\underline{V}_m(A) = \max_z \min_y y^T Az = \frac{2}{5}$. Since $\underline{V}_m(A) = \overline{V}_m(A) = \frac{2}{5}$, these are equilibrium strategies, and

$V_m(A) = \frac{2}{5}$. Therefore, the equilibrium strategies are for the intruder to select each doorway with equal probability $\frac{1}{5}$, and for the guards to cover each doorway with equal probability $\frac{2}{5}$.

This reasoning can be generalized for any m and n, where $m < n$. A is an $n \times \binom{n}{m}$ matrix, and $y^T A$ contains $\sum_{i \in S} y_i$ for every $S \subset \{1, \ldots, n\}$ with m members. Compare to (5.8). Therefore, $\max_z y^T A$ is the sum of the largest m y_i values, and it is minimized by setting every y_i to $\dfrac{1}{n}$. With this choice of y, every element of $y^T A$ is $\dfrac{m}{n}$.

Similarly, Az contains n entries, each of which is a sum of $\binom{n-1}{m-1}$ terms. The ith element is the sum of all the z_j values such that deployment j covers doorway i. Compare to (5.10). Since each deployment covers m doorways, each z_j value appears m times in Az. Therefore, the elements of Az always sum to m. Therefore, $\min_y Az$ can be maximized by setting each element of Az to $\dfrac{m}{n}$. One way to do this is to set $z_j = \dfrac{1}{\binom{n}{m}}$ for every j, as $\binom{n-1}{m-1} \Big/ \binom{n}{m} = \dfrac{m}{n}$. As we saw in (5.12), this is not the only equilibrium strategy for **P2**.

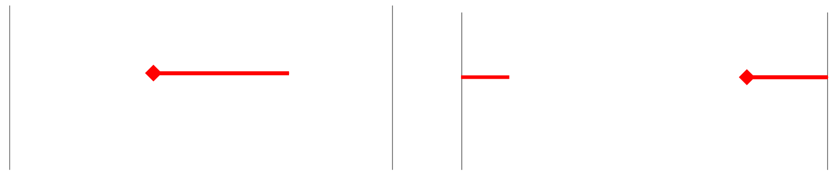
Therefore, $V_m(A) = \underline{V}_m(A) = \overline{V}_m(A) = \frac{m}{n}$. Equilibrium strategies are for the intruder to select each doorway with equal probability $\frac{1}{n}$, and for the guards to cover each doorway with equal probability $\frac{m}{n}$. Notice that the intruder's equilibrium strategy is unchanged by a change in m.

5.2.2 Continuous, vertical corridor

We now consider the case of a vertical corridor of width w, and guards of total length r, which can be divided arbitrarily. **P1** selects vertical paths from $\mathcal{S}_1$ to $\mathcal{S}_2$, and **P2** selects (a) a horizontal line $\mathcal{L}$ that cuts the corridor in two, and (b) an arrangement of guards across $\mathcal{L}$ with total length r. Since the intruder cannot intrude without crossing $\mathcal{L}$, each point in $\mathcal{L}$ acts like a doorway in the discrete examples. Since the intruder must cross every possible $\mathcal{L}$ in order to intrude, the specific choice of $\mathcal{L}$ makes no difference to the game's outcome, provided $\mathcal{L}$ is horizontal, connected to both walls, and of minimal length w. Here both players' choices are continuous.

We can approximate the continuous case with the discrete one by dividing $\mathcal{L}$ into n equal sections, and setting $m = \arg\min_{m\in\mathbb{Z}^+} \left|\frac{m}{n} - \frac{r}{w}\right|$. In other words, select m so that $\frac{m}{n}$ best approximates $\frac{r}{w}$. This m is the integer closest to $\frac{rn}{w}$. According to Section 5.2.1, the coverage value of the discrete example is $\frac{m}{n}$. As n increases, and m accordingly, $\frac{m}{n}$ approaches $\frac{r}{w}$. The difference is always less than $\frac{1}{n}$. This suggests a continuous equilibrium strategy with a value of $\frac{r}{w}$:

- For **P1** select each vertical path with a uniform distribution.
- For **P2** use any strategy where each point of $\mathcal{L}$ is covered with equal proba-



(a) $0 \leq g \leq w - r$ (b) $g > w - r$

Figure 5.2: Using $g \in [0, w]$ to place guards uniformly across a corridor.

bility.

There are many equilibrium strategies for **P2**. Here we give an example of one. Since $\mathcal{L}$ has length w, it has a bijection with $[0, w]$. Select $g \in [0, w]$ with a uniform distribution, and use it to place the guard(s). If the value of g lies in the interval $[0, w - r]$, place one guard across $[g, g + r]$. See Figure 5.2(a). If $g \in (w - r, r]$, place two guards: one across $[g, w]$, and one across $[0, r - w + g]$. See Figure 5.2(b).

We now show that these strategies are equilibrium, independent of the analogous discrete strategies.

Lemma 3: The given strategies for the one-vertical-corridor game are equilibrium strategies.

Proof: Since these strategies guarantee that every point in $\mathcal{L}$ will be covered with a probability of $\frac{r}{w}$, they produce an expected outcome of $\frac{r}{w}$. To show these are equilibrium strategies, we show that $\overline{V}_m = \underline{V}_m = \frac{r}{w}$ using the definitions given in (5.4) and (5.5). To do this we show that for each player, any other strategy produces an inferior worst-case expected outcome. For **P1**, if the distribution is not uniform, there is a region of total length r with a probability higher than $\frac{r}{w}$ of selection. If **P2** places guards exclusively in this region, the expected outcome will be greater than $\frac{r}{w}$. Similarly for **P2**, if the points in $\mathcal{L}$ are not all guarded with the same probability, there are points with a probability lower than $\frac{r}{w}$ of coverage. If **P1** selects this point exclusively to traverse, the expected outcome will be less than $\frac{r}{w}$.

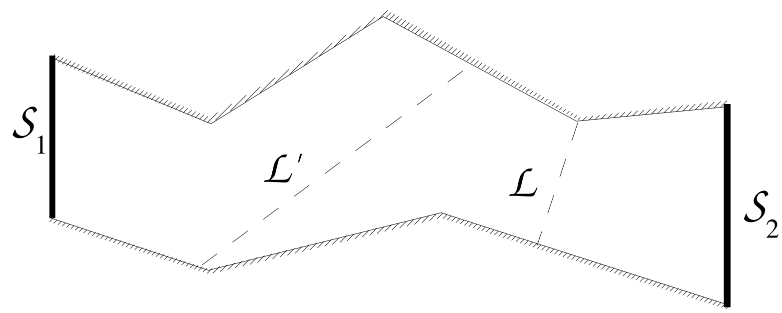


Figure 5.3: Sample single corridor. $\mathcal{L}'$ is an arbitrary barrier, and $\mathcal{L}$ is a minimum barrier.

Therefore, the given strategy is a saddle point, with a value of $\frac{r}{w}$. □

5.2.3 Arbitrary single corridor

We now use the vertical corridor results to construct equilibrium strategies for an arbitrary corridor that is simply connected, but whose width and direction can change arbitrarily. See Figure 5.3 for an example. Let $\mathcal{L}$ be a minimum barrier, which is of length w, and let r be the total guard length, to be distributed by **P2**.

The equilibrium strategies for the arbitrary corridor are:

- For **P1**: select a point in $\mathcal{L}$ with uniform distribution, and select an intrusion path through this point.
- For **P2**: any strategy such that every point in $\mathcal{L}$ is covered with probability $\frac{r}{w}$.

The example **P2** strategy given in Section 5.2.2 can also be used here. The **P1** strategy is not as simple as selecting a point and constructing a path through it. It needs to be a strategy such that for any line segment of length ℓ, the intruder will cross it with a probability of at most $\frac{\ell}{w}$. The details are given in Section 5.3.2. Like the vertical case, this gives a value of $\frac{r}{w}$.

Theorem 2: The given strategies for the one-corridor game are equilibrium strategies, with value $\frac{r}{w}$.

Proof: From Lemma 3, we know that these strategies have an expected outcome of $\frac{r}{w}$. Now suppose **P2** places guards across a different barrier $\mathcal{L}' \neq \mathcal{L}$. Since $\mathcal{L}'$ is not the minimum barrier, it has width $w' > w$. In this case, if **P1** uses the strategy described above (i.e. choosing a crossing point from the uniform distribution on $\mathcal{L}'$), the probability of detection is $\frac{r}{w'} < \frac{r}{w}$. Therefore, choosing any non-minimum $\mathcal{L}$' produces worse outcome for **P2**. Having thus established that **P2**'s strategy must choose $\mathcal{L}$ as the segment on which to place the guards, the remainder of the proof follows directly from Lemma 3.

Since neither player can improve upon the given strategies, they form a saddle point, with expected value $\frac{r}{w}$. □

5.3 General polygonal environments

In this section we use the single-corridor strategies devised in previous sections to find equilibrium strategies for general polygonal environments. In such environments, there may be multiple $\mathcal{S}_i$ components, and multiple homotopy classes of intrusion paths. We can construct equilibrium strategies by viewing the environment as an interconnection of corridors. We use Theorem 2 to determine strategies at a single barrier candidate, and we use the connectivity network $\mathcal{N}$ defined in Section 2.3 to combine these different strategies.

In Section 5.3.1, we give general rules that dictate intruder and guard strategies for polygonal environments. While **P2**'s strategy can be derived directly from these rules, **P1**'s strategy is more complicated; the details are given in Section 5.3.2.

5.3.1 General strategies

We use $\mathcal{G}$ and $\mathcal{N}$ from Section 2.3 to construct strategies for partial coverage in polygonal environments. We make use of the following lemma:

Lemma 4: Any guard deployment that is not restricted to barrier candidates can be replaced by a better one that is across barrier candidates.

Proof: Let $\mathcal{L}$ be one segment across which guards are deployed which is not a barrier candidate. $\mathcal{L}$ must connect two obstacles; otherwise the intruder could avoid $\mathcal{L}$ entirely, and the game would have value 1.

Let b^* be a barrier candidate homotopic to $\mathcal{L}$ that is shorter than $\mathcal{L}$. That b^* exists follows from Theorem 1. The area between $\mathcal{L}$ and b^* forms a corridor. By Theorem 2, placing guards across b^* produces a higher outcome than across $\mathcal{L}$; it is a superior strategy. Therefore, every strategy across every $\mathcal{L}$ that is not a barrier candidate is inferior to a strategy that is across barrier candidates. □

Therefore, we will only consider guards across barrier candidates. Given this constraint, we only consider intruder paths based on which barrier candidates they cross, and where the paths cross a particular barrier candidate.

This game can be viewed as a one-stage game, where **P1** selects an intrusion path through $\mathcal{W}$, and **P2** selects a guard deployment. It can also be viewed as a two-stage game, making use of the barrier candidate graph. In stage one, **P1** selects a path through $\mathcal{N}$ from v_{s_1} to v_{s_2}; his action can be encoded as the set of barrier candidates his path intersects. Similarly, **P2** chooses a guard length r_i for every barrier candidate b_i (each b_i is of length w_i). For this to be a partial coverage problem, the r_i values must sum to a predetermined R, which is strictly less than the length of the minimum barrier, W. In stage 2, **P1** chooses which point in each barrier candidate to cross, and **P2** chooses a guard deployment of total length r_i for each b_i selected in stage 1.

The outcome and strategies of stage 2 can be determined directly from Theorem 2. Suppose **P1** chooses a path through $\mathcal{N}$ and **P2** places guards such that some b_i is the only barrier candidate that is both traversed by the intruder and covered by guards. The union of all faces that correspond to vertices in $\mathcal{N}$ form a corridor, so by

Theorem 2 the expected outcome is $\frac{r_i}{w_i}$. Extending this reasoning to multiple barrier candidates, if **P1** chooses a path that crosses barrier candidates $\{b_1, \ldots, b_K\}$, then the game has outcome $1 - \prod_{i=1}^{K} \left(1 - \frac{r_i}{w_i}\right)$. This expression follows from the fact that the guards only need to detect the intruder once for **P2** to win. The probability of undetected intrusion through b_i is $1 - \frac{r_i}{w_i}$ so the probability of undetected intrusion through all the barrier candidates is $\prod_{i=1}^{K} \left(1 - \frac{r_i}{w_i}\right)$. We denote this value by $A(I, r)$, where $I = \{b_1, \ldots, b_K\}$ is the intruder's path specified in terms of traversed barrier candidates, and $r = (r_1, \ldots, r_N)$ is the guard deployment.

Given this analysis of the second stage, we turn now to the first stage of the game. We build strategies for each player using results from Chapter 2 on minimum complete barriers. **P2**'s equilibrium strategy is based on the minimum complete barrier determined using the method described in Section 2.3. Let B^* be a minimum complete variable-length segment barrier, and denote its total length as W.

P1's equilibrium strategy is based on the maximum flow [23] through $\mathcal{N}$. This flow f assigns a direction and flow value for each edge, such that (1) the flow value is less than or equal to the capacity; and (2) for each vertex except for v_{s_1} and v_{s_2}, the flow in equals the flow out. The minimum cut of a network consists of all the edges where the flow is at capacity. Let $f(b_i)$ be the flow value through b_i in a maximum flow. Since the barrier candidate lengths are used for capacities, b_i has capacity w_i; this means $f(b_i) \leq w_i$. Since the maximum flow is at capacity at the minimum cut, $f(b_i) = w_i$ if $b_i \in B^*$.

Let N be the number of barrier candidates in $\mathcal{N}$. The equilibrium strategies are:

- For **P1**: Select a path so that for each $i = 1, \ldots, N$, b_i is traversed with probability $\frac{f(b_i)}{W}$. For every intrusion path I, the probability that I will be selected is equal to the flow through I divided by W.

- For **P2**: For each $i = 1 \ldots, N$, set $r_i = \frac{w_i}{W} R$ if $b_i \in B^*$, and $r_i = 0$ otherwise.

Place guards of total length r_i across b_i.

Theorem 3: The strategies given above are equilibrium strategies for a polygonal environment, with an expected value of $\frac{R}{W}$.

Proof: First we show that the expected outcome is $\frac{R}{W}$.

Lemma 5: The above strategies give an expected outcome of $\frac{R}{W}$.

Proof: Let E^* be the set of edges in $\mathcal{N}$ corresponding to the barrier candidates in B^*. Divide $\mathcal{N} - E^*$ into two induced subgraphs $\mathcal{N}_1'$ and $\mathcal{N}_2'$. $\mathcal{N}_1'$ contains all vertices reachable from v_{s_1}, and $\mathcal{N}_2'$ contains all vertices reachable from v_{s_2}. In the maximum flow, the flow through edges in E^* are all directed from $\mathcal{N}_1'$ to $\mathcal{N}_2'$.

Therefore, there are no paths through $\mathcal{N}$ of positive flow that traverse E^* multiple times, as this would require moving from $\mathcal{N}_2'$ to $\mathcal{N}_1'$. Therefore, every **P1** path of nonzero probability – being a path with positive flow – traverses the minimum barrier exactly once. If it traverses edge $e_i' \in E^*$, then the expected outcome is $\frac{r_i}{w_i} = \frac{R}{W}$. $\square$

Now we show that the given strategies satisfy (5.6) by showing that each strategy guarantees an expected outcome of $\frac{R}{W}$ or better, even if the other player changes strategies.

First, consider alternative strategies for **P1**. Any path from $\mathcal{S}_1$ to $\mathcal{S}_2$ must cross B^* at least once. If **P2** uses the given strategy, then for any $b_i \in B^*$, $r_i = \frac{w_i}{W}R$. Therefore the resulting outcome will be at least $\frac{r_i}{w_i} = \frac{R}{W}$. This value will be higher if the path crosses B^* multiple times. Therefore, for **P1**, using other strategies produces an equal or higher expected outcome.

Now consider alternative strategies for **P2**. From Lemma 4 we know that a deployment that is not across barrier candidates produces a lower expected outcome. Therefore we write the deployment as $r = (r_1, \ldots, r_N)$, as above. **P2** places guards of total length r_i at barrier candidate b_i.

Given an intruder path I, the probability of being detected by a guard is

$$p_G(I) = 1 - \prod_{\{i|b_i \in I\}} \left(1 - \frac{r_i}{w_i}\right).$$

We can establish bounds on $p_G(I)$, and use these bounds to establish bounds on the expected outcome.

Lemma 6: If $\forall i, 0 \leq r_i \leq w_i$,

$$\prod_{i=1}^{K} \left(1 - \frac{r_i}{w_i}\right) \geq 1 - \sum_{i=1}^{K} \frac{r_i}{w_i}, \tag{5.13}$$

Proof: Proof by induction. (5.13) is trivially true for $K = 1$. Assume true for $K - 1$.

$$\begin{aligned}
\prod_{i=1}^{K} \left(1 - \frac{r_i}{w_i}\right) &= \left(1 - \frac{r_K}{w_K}\right) \prod_{i=1}^{K-1} \left(1 - \frac{r_i}{w_i}\right) \\
&\geq \left(1 - \frac{r_K}{w_K}\right) \left(1 - \sum_{i=1}^{K-1} \frac{r_i}{w_i}\right) \\
&= 1 - \sum_{i=1}^{K} \frac{r_i}{w_i} + \frac{r_K}{w_K} \sum_{i=1}^{K-1} \frac{r_i}{w_i} \\
&\geq 1 - \sum_{i=1}^{K} \frac{r_i}{w_i}
\end{aligned}$$

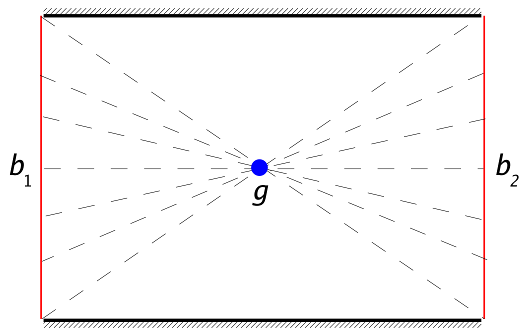
Therefore (5.13) holds. □

Let $p(I)$ be the probability of **P1** selecting I. This is the flow through I divided by W. The expected outcome is therefore

$$\begin{aligned}
\sum_{I} p_G(I)\, p(I) &= \sum_{I} p(I) \left[1 - \prod_{\{i|b_i \in I\}} \left(1 - \frac{r_i}{w_i}\right)\right] \\
&\leq \sum_{I} p(I) \sum_{\{i|b_i \in I\}} \frac{r_i}{w_i} \qquad (5.14) \\
&= \sum_{I} \sum_{\{i|b_i \in I\}} p(I) \frac{r_i}{w_i} \\
&= \sum_{i=1}^{N} \sum_{\{I|b_i \in I\}} p(I) \frac{r_i}{w_i} \\
&= \sum_{i=1}^{N} \frac{r_i}{w_i} \sum_{\{I|b_i \in I\}} p(I) \qquad (5.15) \\
&= \sum_{i=1}^{N} \frac{r_i}{w_i} \frac{f(b_i)}{W} \qquad (5.16) \\
&\leq \sum_{i=1}^{N} \frac{r_i}{W} \qquad (5.17) \\
&= \frac{R}{W}.
\end{aligned}$$

In the inequality above, (5.14) follows from Lemma 6, and (5.15) is a rearrangement of sums. (5.16) follows from the facts that (a) $p(i)$ is the flow through I divided by W, and (b) the flow through a single edge in $\mathcal{N}$ is the sum of the flows through all the paths that contain it. (5.17) follows from $f(b_i) \leq w_i$. Therefore, for **P2**, using other strategies produces an equal or lower expected outcome.

Therefore, if either player changes strategies, the outcome will be equal or worse for that player. Thus the given strategies satisfy (5.6), so they are equilibrium strategies. □



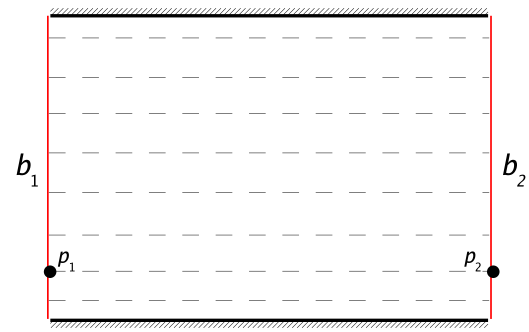


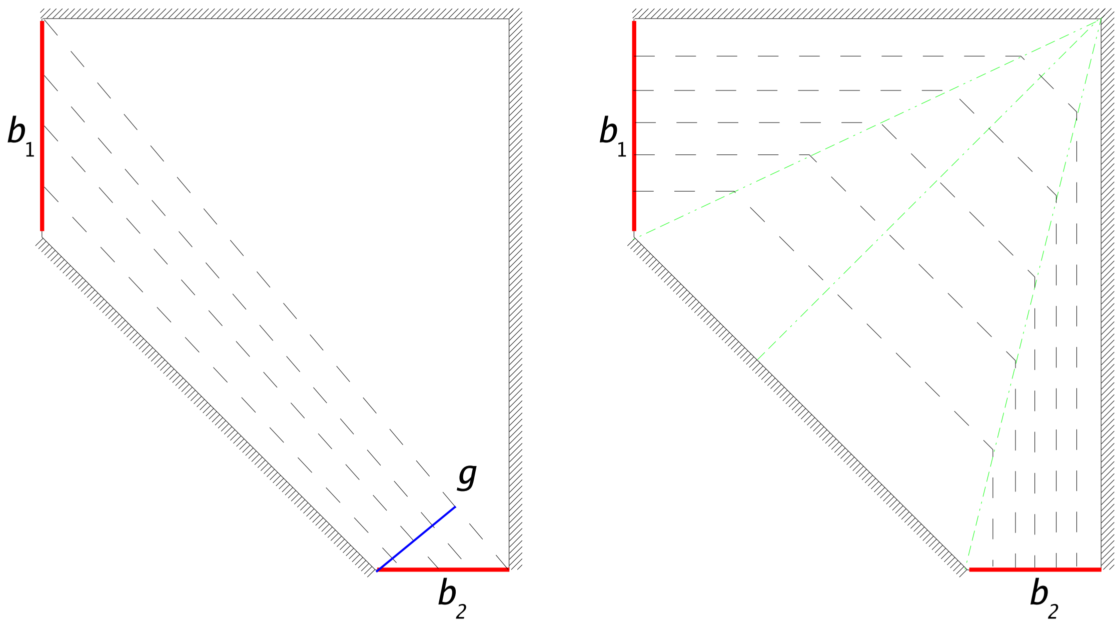
Figure 5.4: Candidate strategies for the intruder traveling from one barrier candidate to another. Dashed lines show example paths.

5.3.2 Intruder strategy details

In this section, we describe the details of **P1**'s equilibrium strategy. The strategy given in Section 5.3.1 only gives strategies for selecting at which points the intruder crosses barrier candidates. It does not state how to travel between barrier candidates. The wrong choice of paths can produce suboptimal results.

For example, consider Figure 5.4(a). While the points intersecting barrier candidates b_1 and b_2 are selected with uniform distributions, all paths can be guarded with a single point guard at g. This clearly produces a much higher expected outcome than the equilibrium value. A preferable strategy is given in Figure 5.4(b). A single value $x \in [0, 1]$ is selected with uniform distribution, and endpoints p_1 and p_2 are the locations of x if $[0, 1]$ were mapped onto b_1 and b_2 respectively. The intruder moves across the line segment connecting p_1 with p_2. Against this **P1** strategy, there is no advantage to placing guards anywhere but b_1 or b_2.

An equilibrium strategy can be constructed by applying this idea to all barrier candidates. Select $x \in [0, 1]$ with uniform distribution, find the appropriate points in all relevant barrier candidates by mapping $[0, 1]$ onto the barrier candidate and recording where x is mapped, and connect the points with lines. Cases where the intruder must chose between multiple barrier candidates are covered later; see Figure



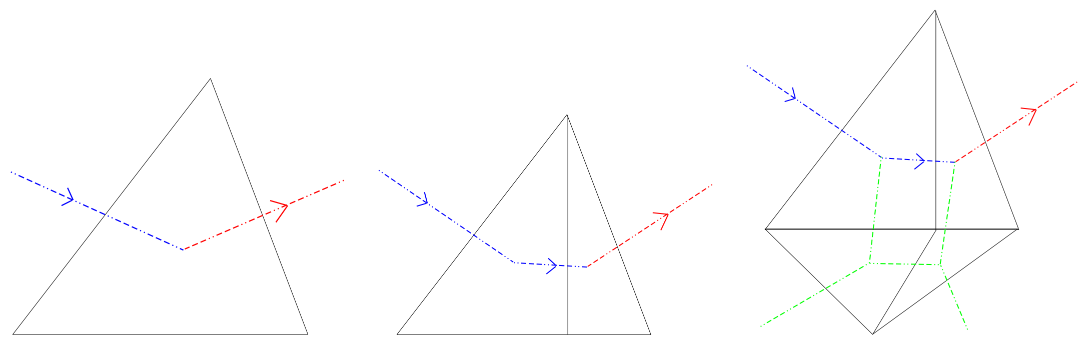
(a) Non-equilibrium strategy (b) Equilibrium strategy

Figure 5.5: Candidate strategies for the intruder traveling between non-parallel barrier candidates. Dot-dashed paths show additional barrier candidates, while dashed lines show example paths.

5.8. Each point in $[0, 1]$ corresponds to a unique intrusion path.

This is not sufficient, as evidenced by the example in Figure 5.5(a). Guard g clearly covers all paths, even though it is shorter than b_1 or b_2. The problem is neither the selection of x, the choice of points in barrier candidates, nor the connections between them. Rather, the intruder does not make adequate use of the polygon. The desired strategy is shown in Figure 5.5(b). Notice the absence of untraversed regions. This strategy is achieved by (a) triangulating non-triangular faces, and (b) adding altitudes to appropriate triangles. More specifically, for any triangular face with one in edge and one out edge, add the altitude to the edge with no flow. If there is an adjacent triangular face, add an edge to the opposite vertex. See Figure 5.6.

Continue dividing triangles until there are no such triangles. All triangular faces now either (1) have no flow, (2) are right triangles with one zero-flow leg, or (3)

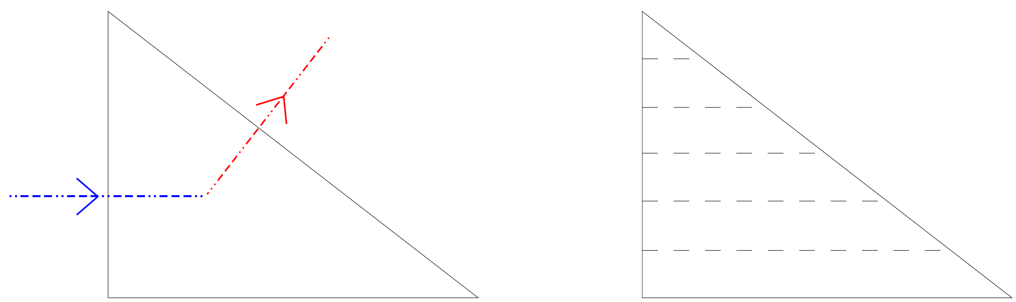


(a) Before

(b) After,
without adjacent face

(c) After,
with adjacent face.

Figure 5.6: Transforming a triangle with one 0-flow edge. Barrier candidates are black and solid; network edges are colored and dot-dashed.



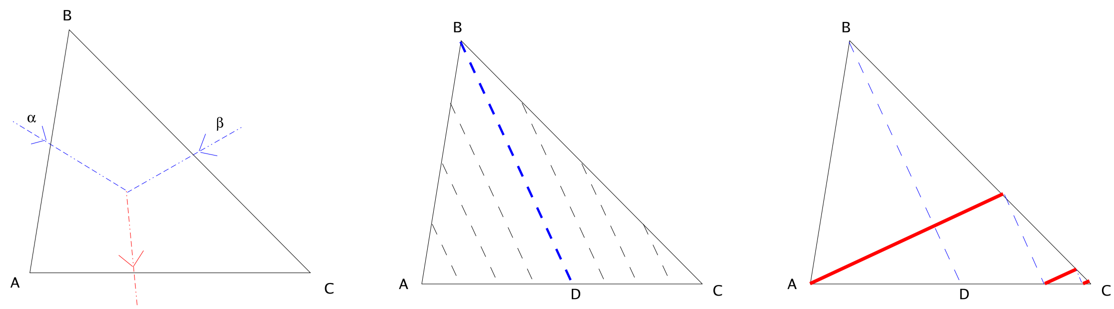
(a) Triangle, with flows

(b) Example intruder paths

Figure 5.7: Intruder strategy across a right triangle with a zero-flow leg. Barrier candidates are black and solid, network edges are colored and dot-dashed, and intruder paths are dashed.

have flow through all three legs. The intruder will not travel through faces of type (1). When moving through type (2), he will use paths parallel to the zero-flow leg. See Figure 5.7. A face of type (3) has either 2 in edges and 1 out, or vice versa. In Figure 5.8(a), triangle ABC has input flow of α through $\overline{AB}$, input flow of β through BC, and output flow of $\alpha + \beta$ through $\overline{AC}$. The intruder will use paths parallel to $\overline{BD}$, where D is selected so that $\frac{AD}{DC} = \frac{\alpha}{\beta}$. Figure 5.8(b) shows these paths. If there are two edges flowing out and one in, simply reverse the directions from Figure 5.8(b).

In the third case, the shortest guard that covers every path is not AC, but a



(a) Triangle, with flows (b) Example intruder paths (c) Minimum guard deployment (thick red).

Figure 5.8: Intruder strategy across a triangle with three nonzero-flow edges. Barrier candidates are black and solid, network edges are colored and dot-dashed, and intruder paths are dashed.

set of guards perpendicular to AD, shown in Figure 5.8(3). The total length is $AC \sin\theta$, where θ is the measure of $\angle ADB$. As long as this value is no less than $\alpha + \beta$, this strategy is still equilibrium, as this guard is not an improvement over the minimum barrier. If AC is part of the minimum barrier, then $\theta = \frac{\pi}{2}$ is forced, meaning $\overline{BD} \perp \overline{AC}$. In this case, $\overline{BD}$ is added to the connectivity network with a capacity of 0. This does not decrease the minimum barrier, as any barrier containing these 0-capacity edges involves a trapezoid like in Figure 5.9. In the figure, z_0 and z_1 are new 0-capacity edges, both of which are perpendicular to b^*. It is impossible for b' to be shorter than b^*. Therefore any such b' cannot be in the minimum barrier; its corresponding edge cannot be in the minimum cut. Since any cut that does not contain 0-capacity edges is a barrier in the original connectivity network, it cannot be the minimum. Therefore, adding these edges does not change the minimum cut.

This method produces equilibrium strategies for **P1**. The intruder selects $x \in [0, 1]$ with uniform distribution, and uses it to select a point along the edge of $\mathcal{S}_1$. The intruder will then follow the given paths through the triangles of the extended barrier candidate graph until arriving at $\mathcal{S}_2$.

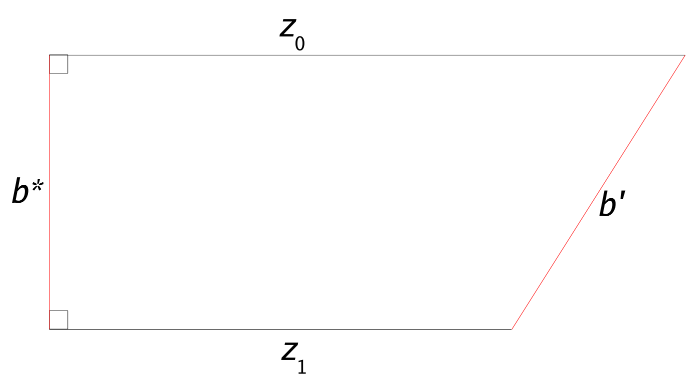


Figure 5.9: Trapezoid generated by exploiting 0-capacity edges z_0 and z_1. This shows that the minimum cut will contain b^* and not b'.

Chapter 6

Partial coverage for fixed-length guards

In this chapter, we give partial coverage values and strategies for fixed-length guards. We give methods for both segment and circle guards. We will show how the methods relate to two NP-Complete problems: minimum fixed-length complete barrier, described in Chapters 3 and 4, and maximum thick paths [42].

In Section 6.1 we give strategies for one discretized corridor. We use this in Section 6.2 to build continuous single-corridor strategies. In Section 6.3 we use the single-corridor strategies to motivate strategies for general environments. We give strategies for circle guards in Section 6.3.1, and derive segment guard strategies from them in Section 6.3.2.

6.1 One discrete corridor

In this section, we give strategies for a single vertical corridor, with $\mathcal{S}_1$ at the bottom and $\mathcal{S}_2$ at the top. We model the corridor as a discrete set of doorways, like the examples in Section 5.2.1. Like Section 5.2.1, we will use the discrete notation of Section 5.1.1, before moving to general notation in the rest of the chapter. In the examples in this section, a single guard can see ρ consecutive doorways. In this section, we will show that, for n doorways, and m guards, each of which can see ρ consecutive doorways, the equilibrium strategies are:

- For **P1**:

1. Select $\ell = \left\lceil \frac{n}{\rho} \right\rceil$ clusters of doorways, such that no two doorways from different clusters are less than ρ doors apart.

2. Select each cluster with probability $\frac{1}{\ell}$, and choose any doorway in the cluster to traverse.

- For **P2**:

1. Divide the line of doorways into ℓ contiguous sections, each of which is at most ρ doorways wide.

2. Select m sections, so that each section is selected with probability $\frac{m}{\ell}$, and choose any set of guards that cover these m sections.

Using these strategies gives a partial coverage value of $V_m = \frac{m}{\left\lceil \frac{n}{\rho} \right\rceil}$. This value is the same for any ρ, ρ' where $\left\lceil \frac{n}{\rho} \right\rceil = \left\lceil \frac{n}{\rho'} \right\rceil$, even if $\rho' \neq \rho$. This has the nonobvious consequence that a small change in the guard range may produce no change in the partial coverage value.

To find these strategies, we make use of the concept of domination in noncooperative games. We show that some actions are worse than others, and can be removed without changing the equilibrium strategies. Removing these actions makes for a simpler game to analyze. Though we use the discrete notation in this section, the concept applies equally to continuous games.

In this section, as in Section 5.1.1, we represent the game with the matrix A, where a_{ij} is the outcome of the game when **P1** plays action i and **P2** plays action j. Each **P1** action corresponds to a row i in A, and each **P2** action corresponds to a column j in A.

Definition: **P1** action i_1 *dominates* action i_2 if $a_{i_1,j} \leq a_{i_2,j}$ for every **P2** action j, with strict inequality for at least one j. Similarly, **P2** action j_1 *dominates* action j_2 if $a_{i,j_1} \geq a_{i,j_2}$ for every **P1** action i, with strict inequality for at least one i.

If i_1 dominates i_2, then **P1** has nothing to gain by playing i_2, which never produces a better outcome than i_1, and sometimes produces a worse one. Therefore, for fixed strategies **P1** should never play i_2. For mixed strategies denoted by $y = [y_1, \ldots, y_m]^T$, where **P1** plays action i with probability y_i, any strategy y where $y_{i_2} > 0$ can be replaced by a strategy y' where $y'_{i_2} = 0$, $y'_{i_1} = y_{i_1} + y_{i_2}$, and $y'_i = y_i$ for all other i. In other words, y' plays i_1 instead of i_2. The new strategy y' will produce an equal or better value for **P1**, regardless of **P2**'s strategy, even if y is an equilibrium strategy. Therefore A has at least one saddle point that does not use row i_2. The same reasoning can be applied to **P2**'s strategy.

By eliminating these dominated options from A, and giving them 0 probabilities in the computed strategies, we will be left with a smaller matrix, but with at least one saddle point remaining. Since all saddle points have the same value, removing saddle points does not change the value. Removing dominated rows and columns thus leaves us with a smaller matrix in which it is easier to find equilibrium strategies, without changing the game's value or losing all equilibrium strategies.

In this section, we derive the above strategies by first deriving them for a single guard, in Section 6.1.1, and then extending this to multiple guards, in Section 6.1.2.

6.1.1 One guard

In this section we consider the case where there are n doorways, and one guard that can see ρ consecutive doorways. Consider the example shown in Figure 6.1, in which $n = 5$ and $\rho = 2$. We will first demonstrate how to use domination to determine the equilibrium strategies for this specific example. We then proceed to the general case, for any n and ρ. We will perform a series of dominations, alternating between blocks of rows and blocks of columns, until no more are possible. Lemmas 7 and 8 give rules for dominating rows, and Lemmas 9 through 13 give rules for dominating columns. This reduces A to a smaller A', from which we determine equilibrium

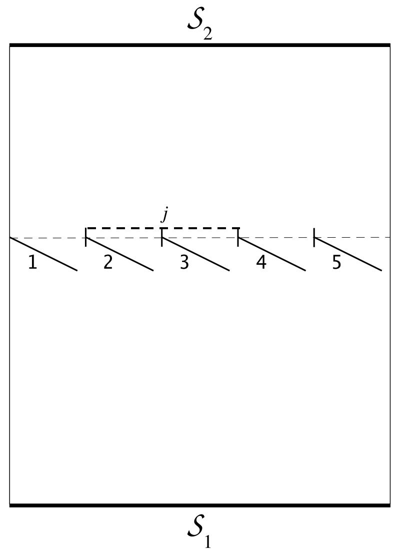


Figure 6.1: Discretized corridor

strategies in Theorem 4.

We describe the example in Figure 6.1 as a matrix $A = \{a_{ij}\}$, where a_{ij} is 1 if the guard covering doorways j and $j+1$ detects an intruder going through doorway i, and 0 otherwise. Equivalently, $a_{ij} = 1$ if $i \in \{j, j+1\}$, and $a_{ij} = 0$ otherwise. Therefore,

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.1}$$

As in section 5.2.1, $\overline{V} = 1$ and $\underline{V} = 0$ for pure strategies, so there are no pure equilibrium strategies. Therefore we look at mixed strategies y and z.

We can reduce A by exploiting domination between rows. In this example, $a_{1j} \leq a_{2j}$ for every j , with strict inequality at $j = 2$; therefore row 1 dominates row 2. Similarly, since $a_{5j} \leq a_{4j}$, with strict inequality at $j = 4$, row 5 dominates row 4.

Therefore we eliminate rows 2 and 4, producing a reduced game matrix:

$$A' = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.2}$$

We also have a reduced **P1** strategy $y' = (y_1, y_3, y_5)^T$. Since $y'^T A' = (y_1, y_3, y_3, y_5)$, **P2** will maximize the outcome by playing the column with the maximum y_i value. Therefore, $\max\limits_z y'^T A' z = \max(y_1, y_3, y_5)$. Since $y_1 + y_3 + y_5 = 1$, this value can be minimized by setting $y_1 = y_3 = y_5 = \frac{1}{3}$. Therefore
$\overline{V}_m(A') = \min\limits_y \max\limits_z y'^T A' z = \frac{1}{3}$.

Similarly, $A'z = (z_1, z_2 + z_3, z_4)^T$, and **P1** can minimize this by playing the row with minimum value. Therefore, $\min\limits_y y'^T A' z = \min(z_1, z_2 + z_3, z_4)$. Since $z_1 + z_2 + z_3 + z_4 = 1$, this value can be maximized by setting $z_1 = z_2 + z_3 = z_4 = \frac{1}{3}$. Therefore $\underline{V}_m(A') = \max\limits_z \min\limits_y y'^T A' z = \frac{1}{3}$.

Since $\underline{V}_m(A) = \overline{V}_m(A) = \frac{1}{3}$, $y'^* = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)^T$ and $z^* = \left(\frac{1}{3}, z_2, \frac{1}{3} - z_2, \frac{1}{3}\right)^T$ are equilibrium strategies for any $z_2 \in \left[0, \frac{1}{3}\right]$, and $V_m(A) = \frac{1}{3}$. Returning to the original matrix, the equilibrium strategies are $y^* = \left(\frac{1}{3}, 0, \frac{1}{3}, 0, \frac{1}{3}\right)^T$, and $z^* = \left(\frac{1}{3}, z_2, \frac{1}{3} - z_2, \frac{1}{3}\right)^T$, for any $0 \le z_2 \le \frac{1}{3}$. Therefore, the equilibrium strategies are for the intruder to go through the far left, middle, and far right doorways with equal probability, and for the guard to cover each of those doorways with equal probability.

If we were to seek equilibrium strategies for this matrix without removing rows, we would still arrive at $y^* = \left(\frac{1}{3}, 0, \frac{1}{3}, 0, \frac{1}{3}\right)^T$, and $z^* = \left(\frac{1}{3}, z_2, \frac{1}{3} - z_2, \frac{1}{3}\right)^T$. In this case domination preserved the game's equilibria. However, in later cases, removing dominated rows and columns will remove equilibrium strategies. This will not change the value of the game, nor undermine the calculated equilibrium strategies, as we will see later.

Now we consider the game for any n and ρ, where $\rho < n$. We will first describe the outcome matrix A. We will then use domination to reduce A to a smaller A' from which it is straightforward to determine equilibrium strategies and the game's value.

We define the outcome matrix A to be a $n \times (n - \rho + 1)$ matrix, where a_{ij} represents **P1** selecting doorway i, and **P2** covering doorways $j, j+1, \ldots, j+\rho-1$. Therefore, $a_{ij} = 1$ if $j \leq i \leq j + \rho - 1$, and $a_{ij} = 0$ otherwise. Equivalently, $a_{ij} = 1$ if $i - \rho + 1 \leq j \leq i$, and $a_{ij} = 0$ otherwise. This gives a matrix of the form

$$A = \begin{bmatrix} 1 & 0 & 0 & & & \cdots & & 0 \\ 1 & 1 & 0 & & & \cdots & & 0 \\ \vdots & & \ddots & \ddots & & & & \vdots \\ 1 & & \cdots & 1 & 0 & & \cdots & 0 \\ 0 & 1 & \cdots & & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 & & \cdots & 1 & 0 \\ 0 & \cdots & & 0 & 1 & \cdots & & 1 \\ \vdots & & & & \ddots & \ddots & & \vdots \\ 0 & & \cdots & & & 0 & 1 & 1 \\ 0 & & \cdots & & & 0 & 0 & 1 \end{bmatrix}. \tag{6.3}$$

A is a band diagonal matrix with lower bandwidth $\rho - 1$ and upper bandwidth 0, with all bands consisting entirely of 1's. Alternatively, all diagonals of length $n - \rho + 1$ contain all 1's. The remaining diagonals contain all 0's.

We apply a series of dominations to A, similar to the dominations used in the specific example above. This time, there are multiple rounds of domination, as removing a set of dominated rows can create new column dominations, and vice versa.

Lemma 7: Row 1 dominates every row of A in $\{2, \ldots, \rho\}$.

Proof: Since the only guard deployment j that can cover doorway 1 is $j = 1$, $a_{11} = 1$, and $a_{12} = a_{13} = \cdots = a_{1n} = 0$. For $i \in \{2, \ldots, \rho, \}$, doorway i is covered by $j = 1$, but also by $j = 2, \ldots, i$. Therefore,

$a_{i1} = a_{i2} = \cdots = a_{ii} = 1$, and $a_{i,i+1} = a_{i,i+2} = \cdots = a_{in} = 0$. It follows that for every $i \in \{2, \ldots, \rho\}$, $a_{1j} \leq a_{ij}$ for every j, with strict inequality when $2 \leq j \leq i$. Therefore, $i = 1$ dominates $i = 2, \ldots, \rho$. $\square$

Lemma 8: Row n dominates every row of A in $\{n - \rho + 1, \ldots, n - 1\}$.

Proof: Since the only guard deployment j that can cover doorway n is $j = n-\rho+1$, $a_{n,n-\rho+1} = 1$, and $a_{n1} = a_{n2} = \cdots = a_{n,n-\rho} = 0$. For $i \in \{n - \rho + 1, \ldots, n - 1, \}$, doorway i is covered by $j = n-\rho+1$, but also by $j = i-\rho+1, \ldots, n-\rho$. Therefore, $a_{i,i-\rho+1} = a_{i,i-\rho+2} = \cdots = a_{i,n-\rho+1} = 1$, and $a_{i1} = a_{i2} = \cdots = a_{i,i-\rho} = 0$. It follows that for every $i \in \{n - \rho + 1, \ldots, n - 1\}$, $a_{nj} \leq a_{ij}$ for every j, with strict inequality when $i-\rho+1 \leq j \leq n-\rho$. Therefore, $i = n$ dominates $i = n-\rho+1, \ldots, n-1$. $\square$

Using lemmas 7 and 8, we can remove $2\rho - 2$ rows from A, producing a new matrix A', which is $(n - 2\rho + 2) \times (n - \rho + 1)$ (except if $n \leq 2\rho$; see below). How we proceed with additional dominations depends on the relationship between n and ρ. Either $n \leq 2\rho$, $2\rho < n \leq 3\rho$, or $n > 3\rho$. In each one of these three cases, the domination proceeds differently.

Case 1: $n \leq 2\rho$. In this case, there are no rows between the largest i dominated by $i = 1$, ρ, and the smallest i dominated by $i = n$, $n - \rho + 1$. This means that every row between 2 and $n - 1$ has been dominated. Therefore there are only two rows in A'.

Lemma 9: If $n \leq 2\rho$, column 1 dominates every column of A' in $\{2, \ldots, n - \rho\}$.

Proof: In this case, every column of A' except for $j = 1$ and $j = n-\rho+1$ consists only of 0's. This is because if $2 \leq j \leq n-\rho+1$, the guard deployed at j does not see doorways 1 or n, which are the only undominated options for **P1**. Therefore, for every $j \in \{2, \ldots, n-\rho\}$, $a_{i1} \geq a_{ij}$ for every i, with strict inequality at $i=1$. It follows that column 1 dominates every column in $\{2, \ldots, n-\rho\}$[1]. $\square$

After removing these columns, we are left with two undominated rows (1 and n), and 2 undominated columns (1 and $n-\rho+1$). $j=1$ sees $i=1$, and $j=n-\rho+1$ sees $i=n$, but not vice versa. Therefore, A' is the 2×2 identity matrix I_2. The domination process stops here. We call this *Terminating condition 1*, and we will return to it later.

Case 2: $2\rho < n \leq 3\rho$. In this case, A' contains a rectangular block of 1's: $a_{ij} = 1$ for every $i \in \{\rho+1, \ldots, n-\rho\}$ and every $j \in \{n-2\rho+1, \ldots, \rho+1\}$. These correspond to the rows between the largest i dominated by $i=1$, ρ, and the smallest i dominated by $i=n$, $n-\rho+1$. Since $n \leq 3\rho$, there are at most ρ of these, so they can all be covered by a single guard j, as long as $n-2\rho+1 \leq j \leq \rho+1$.

Lemma 10: If $2\rho < n \leq 3\rho$, column $\rho+1$ dominates every column of A' in $\{2, \ldots, n-2\rho\}$.

Proof: For these columns, $a_{ij} = 1$ iff $j \leq i \leq j+\rho-1$. Since $n \leq 3\rho$ implies $j \leq n-2\rho \leq \rho$, the undominated rows where $a_{ij} = 1$ are $i \in \{\rho+1, \ldots, j+\rho-1\}$. But we saw before that $a_{i,\rho+1} = 1$ for all $i \in \{\rho+1, \ldots, n-1\}$. Therefore, for any $j \in \{2, \ldots, n-2\rho\}$, $a_{i,\rho+1} \geq a_{ij}$ for all undominated i, with strict inequality when $j+\rho \leq i \leq n-\rho$. We know there is at least one such i since $j \leq n-2\rho$ implies $j+\rho \leq n-\rho$. Therefore, $j=\rho+1$ dominates every $j \in \{2, \ldots, n-2\rho\}$. $\square$

Lemma 11: If $2\rho < n \leq 3\rho$, column $\rho+1$ dominates every column of A' in $\{\rho+2, \ldots, n-\rho\}$.

Proof: For these columns, $a_{ij} = 1$ iff $j \leq i \leq j+\rho-1$. Since $n \leq 3\rho$ im-

[1] One could similarly show that column $n-\rho+1$ dominates all these columns.

plies $j + \rho - 1 \geq 2\rho + 1 \geq n - \rho + 1$, the undominated rows where $a_{ij} = 1$ are $i \in \{j, \ldots, n - \rho\}$. But we saw before that $a_{i,\rho+1} = 1$ for all $i \in \{\rho + 1, \ldots, n - 1\}$. Therefore, for any $j \in \{\rho + 2, \ldots, n - \rho\}$, $a_{i,\rho+1} \geq a_{ij}$ for all undominated i, with strict inequality when $\rho + 1 \leq i \leq j - 1$. We know there is at least one such i since $j \geq \rho + 2$ implies $j - 1 \geq \rho + 1$. Therefore, $j = \rho + 1$ dominates every $j \in \{\rho + 2, \ldots, n - \rho\}$. $\square$

Note that in both Lemmas 10 and 11, $\rho + 1$ can be replaced by any column j where $n - 2\rho + 1 \leq j \leq \rho + 1$.

At this point, the dominations have eliminated all the rows and columns except for (a) the first and last row, (b) the first and last column, and (c) the rows and columns that produce the rectangular block of 1's. There are $R = n - 2\rho$ such rows and $C = 3\rho - n + 1$ such columns in the middle. The resulting matrix is:

$$A' = \begin{bmatrix} 1 & 0_{1\times C} & 0 \\ 0_{R\times 1} & 1_{R\times C} & 0_{R\times 1} \\ 0 & 0_{1\times C} & 1 \end{bmatrix}, \tag{6.4}$$

where $1_{a\times b}$ and $0_{a\times b}$ are the $a \times b$ matrices where every element of the matrix is 1 and 0 respectively. It is evident that there are no more dominating rows or columns. This is *Terminating Condition 2*, which we will also return to later.

Case 3: $n > 3\rho$.

Lemma 12: If $n > 3\rho$, column $\rho + 1$ dominates every column of A' in $\{2, \ldots, \rho\}$.

Proof: For any $j \in \{2, \ldots, \rho\}$, the rows where $a_{ij} = 1$ are $i \in \{j, \ldots, j + \rho - 1\}$. Since $j \leq \rho$, the undominated rows among these are $\{\rho + 1, j + \rho - 1\}$. However, the rows where $a_{i,\rho+1} = 1$ are $i \in \{\rho + 1, 2\rho\}$, none of which are dominated. Therefore, if $2 \leq j \leq \rho$, then $a_{i,\rho+1} \geq a_{ij}$ for all i, with strict inequality if $j + \rho \leq i \leq 2\rho$. There is at least one i since $j \leq \rho$ implies $j + \rho \leq 2\rho$. Therefore $j = \rho + 1$ dominates every $j \in \{2, \ldots, \rho\}$. $\square$

Lemma 13: Column $n-2\rho+1$ dominates every column of A' in $\{n-2\rho+2, \ldots, n-\rho\}$.

Proof: For any $j \in \{n-2\rho+2, \ldots, n-\rho\}$, the rows where $a_{ij} = 1$ are $i \in \{j, \ldots, j+\rho-1\}$. Since $j \geq n-2\rho+2$, the undominated rows among these are $\{j, n-\rho\}$. However, the rows where $a_{i,n-2\rho+1} = 1$ are $i \in \{n-2\rho+1, n-\rho\}$, none of which are dominated. Therefore, if $n-2\rho+2 \leq j \leq n-\rho$, then $a_{i,n-2\rho+1} \geq a_{ij}$ for all i, with strict inequality if $n-2\rho+1 \leq i \leq j-1$. There is at least one i since $j \geq n-2\rho+2$ implies $j-1 \geq n-2\rho+1$. Therefore, $j = n-2\rho+1$ dominates every $j \in \{n-2\rho+2, \ldots, n-\rho\}$. $\square$

After applying lemmas 7, 8, 12, and 13, the undominated actions for the players are:

$$\begin{aligned} i &\in \{1\} \cup \{\rho+1, \ldots, n-\rho\} \cup \{n\} \\ j &\in \{1\} \cup \{\rho+1, \ldots, n-2\rho+1\} \cup \{n-\rho+1\}, \end{aligned}$$

and the reduced outcome matrix is

$$A' = \begin{bmatrix} 1 & 0_{1\times(n-3\rho+1)} & 0 \\ 0_{(n-2\rho)\times 1} & A_{(n-2\rho)\times(n-3\rho+1)} & 0_{(n-2\rho)\times 1} \\ 0 & 0_{1\times(n-3\rho+1)} & 1 \end{bmatrix}.$$

A' contains the outcome matrix for the game of a corridor with $n-2\rho$ doors, and one guard with length ρ. The first and last rows cannot dominate or be dominated by any other rows in A'; the same can be said about the first and last columns. Therefore, we can apply lemmas 7, 8, 12, and 13 to the middle matrix, multiple times. This third case is not a terminating condition like the others; we can repeat this process of removing $2\rho-2$ dominated rows from each set until the middle matrix reaches case 1 or 2. After s iterations, the resulting sets are:

$$\begin{aligned} i &\in \{k\rho+1\}_{k=0}^{s-1} \cup \{s\rho+1,\ldots,n-s\rho\} \cup \{n-k\rho\}_{k=0}^{s-1} \\ j &\in \{k\rho+1\}_{k=0}^{s-1} \cup \{s\rho+1,\ldots,n-(s+1)\rho+1\} \cup \{n-(k+1)\rho+1\}_{k=0}^{s-1}, \end{aligned}$$

and the resulting reduced outcome matrix is

$$A' = \begin{bmatrix} I_s & 0_{s\times C} & 0_{s\times s} \\ 0_{R\times s} & A_{R\times C} & 0_{R\times s} \\ 0_{s\times s} & 0_{s\times C} & I_s \end{bmatrix}, \tag{6.5}$$

where $R = n - 2s\rho$, and $C = n - (2s+1)\rho + 1$.

The middle matrix reaches a terminating condition when $R \leq 3\rho$, i.e. $s \geq \frac{n}{2\rho} - \frac{3}{2}$. Therefore, set $s = \lceil \frac{n}{2\rho} - \frac{1}{2} \rceil - 1$. Furthermore, if $R \leq 2\rho$, then the process will reach Terminating Condition 1, leaving A' as an identity matrix. This happens when $n - 2s\rho \leq 2\rho$, i.e. $s \geq \frac{n}{2\rho} - 1$. Otherwise, it will reach Terminating Condition 2.

Therefore, which terminating condition the domination process will reach depends on whether
$\lceil \frac{n}{2\rho} \rceil = \lceil \frac{n}{2\rho} - \frac{1}{2} \rceil$.

Lemma 14: $\forall n, \rho \in \mathbb{Z}^+$, $\lceil \frac{n}{2\rho} \rceil = \lceil \frac{n}{2\rho} - \frac{1}{2} \rceil$ iff $\lceil \frac{n}{\rho} \rceil$ is even.

Proof: If $\lceil \frac{n}{\rho} \rceil$ is even, then $\frac{n}{\rho} = 2i - f$ for some $i \in \mathbb{Z}$ and $f \in [0,1)$. Therefore,

$$\begin{aligned} \left\lceil \frac{n}{2\rho} \right\rceil &= \left\lceil i - \frac{f}{2} \right\rceil = i \\ \left\lceil \frac{n}{2\rho} - \frac{1}{2} \right\rceil &= \left\lceil i - \frac{f}{2} - \frac{1}{2} \right\rceil = i = \left\lceil \frac{n}{2\rho} \right\rceil, \end{aligned}$$

since $\frac{f}{2} \in \left[0, \frac{1}{2}\right)$.

Similarly, if $\lceil \frac{n}{\rho} \rceil$ is odd, then $\frac{n}{\rho} = 2i + 1 - f$ for some $i \in \mathbb{Z}$ and $f \in [0, 1)$. Therefore,

$$\begin{aligned} \left\lceil \frac{n}{2\rho} \right\rceil &= \left\lceil i + \frac{1}{2} - \frac{f}{2} \right\rceil = i + 1 \\ \left\lceil \frac{n}{2\rho} - \frac{1}{2} \right\rceil &= \left\lceil i - \frac{f}{2} \right\rceil = i \neq \left\lceil \frac{n}{2\rho} \right\rceil . \end{aligned}$$

since $\frac{f}{2} \in \left[0, \frac{1}{2}\right)$.

Therefore, $\lceil \frac{n}{2\rho} \rceil = \lceil \frac{n}{2\rho} - \frac{1}{2} \rceil$ iff $\lceil \frac{n}{\rho} \rceil$ is even. $\square$

We can use all these lemmas together to find $V_m(A)$.

Theorem 4: The discrete one-corridor game with n doorways and one guard of width ρ has value $V_m(A) = \left\lceil \frac{n}{\rho} \right\rceil^{-1}$.

Proof: Apply lemmas 7, 8, 12, and 13 repeatedly, to reach a terminating condition. Which condition is reached depends on whether $\ell = \lceil \frac{n}{\rho} \rceil$ is even or odd (Lemma 14).

If ℓ is even, the process ends with Terminating Condition 1. Applying lemma 9 reduces A' to I_ℓ, the $\ell \times \ell$ identity matrix. The undominated actions for both players are:

$$\begin{aligned} i &\in \{k\rho + 1\}_{k=0}^{\ell/2} \cup \{n - k\rho\}_{k=0}^{\ell/2} \\ j &\in \{k\rho + 1\}_{k=0}^{\ell/2} \cup \{n - (k+1)\rho + 1\}_{k=0}^{\ell/2} . \end{aligned}$$

The equilibrium strategies are $y'_i = z'_j = \frac{1}{\ell}$ for all i, j, and $V_m(A') = \frac{1}{\ell}$.

If ℓ is odd, the process ends with Terminating Condition 2, and $s = \frac{\ell - 1}{2}$. Apply lemmas 10 and 11. The resulting reduced outcome matrix is

$$A' = \begin{bmatrix} I_{s+1} & 0_{s+1\times C} & 0_{s+1\times s+1} \\ 0_{R\times s+1} & 1_{R\times C} & 0_{R\times s+1} \\ 0_{s+1\times s+1} & 0_{s+1\times C} & I_{s+1} \end{bmatrix}, \tag{6.6}$$

where

$$\begin{aligned} R &= n - 2\,(s+1)\,\rho \\ C &= (2s+3)\,\rho - n + 1. \end{aligned}$$

The undominated actions for both players are

$$\begin{aligned} i &\in \{k\rho+1\}_{k=0}^{s} \cup \{(s+1)\,\rho+1,\ldots,n-(s+1)\,\rho\} \cup \{n-k\rho\}_{k=0}^{s} \\ j &\in \{k\rho+1\}_{k=0}^{s} \cup \{(s+1)\,\rho+1,\ldots,n-(s+2)\,\rho+1\} \cup \{n-(k+1)\,\rho+1\}_{k=0}^{s}. \end{aligned}$$

To obtain equilibrium strategies for this A', set the y_i' and z_j' values for the rows and columns of the I_{s+1} portions of A' to $\frac{1}{\ell}$. These are the rows $i \in \{1, 2, \ldots, s+1,$ $R+s+2, R+s+3, \ldots, R+2s+2\}$, and the columns $j \in \{1, 2, \ldots, s+1,\ C+s$ $+2, C+s+3, \ldots, C+2s+2\}$. For the remaining rows $i \in \{s+2, \ldots, R+s+1\}$ and $j \in \{s+2, \ldots, C+s+1\}$ any y_i' and z_j' values such that $\sum_{i=s+2}^{R+s+1} y_i' = \sum_{j=s+2}^{C+s+1} z_j' = \frac{1}{\ell}$ form equilibrium strategies. In other words, each maximal $1_{a\times b}$ block in A' should be selected by each player with probability $\frac{1}{\ell}$. Therefore $V_m(A') = \frac{1}{\ell}$.

In both cases, $V_m(A) = V_m(A') = \frac{1}{\ell} = \left\lceil \frac{n}{\rho} \right\rceil^{-1}$, and the equilibrium strategies are to select each maximal rectangular block of 1's with probability $\left\lceil \frac{n}{\rho} \right\rceil^{-1}$. □

Here are some consequences of the above derivations:

- The value of the game does not depend on the terminating condition.
- If $\rho = 1$, $A' = A = I_n$, regardless of whether n is even or odd.
- Since V_m is defined by a ceiling function, in many cases increasing ρ does not increase the coverage value. Only a change in ρ that changes ℓ will change V_m.

This strategy is not unique. This is because removing dominated rows and columns can remove saddle points. Furthermore, the dominated rows and columns can be removed in a different order, resulting in different removed rows, and therefore different strategies. For example, in the sequence of dominations used above, the high-numbered actions and low-numbered actions were dominated in the same iterations. It is possible to only remove low-numbered actions. This would produce a different A', and consequently a different saddle point. However, this multitude of strategies does not change the game's value, as (1) all saddle points achieve the same value, and (2) if **P1** selects a strategy from one saddle point, and **P2** selects a strategy from a different saddle point, the resulting strategy pair is also a saddle point [2]. Therefore, any equilibrium strategy derived from any domination process is an equilibrium, regardless of how the other player derives his strategy.

We can use these additional saddle points to construct a more general equilibrium strategy:

- For **P1**:
  1. Select $\ell = \left\lceil \frac{n}{\rho} \right\rceil$ clusters of doorways, such that no two doorways from different clusters are less than ρ doors apart.
  2. Select each cluster with probability $\frac{1}{\ell}$, and choose any doorway in the cluster to traverse.

- For **P2**:

  1. Divide the line of doorways into ℓ contiguous sections, each of which contains at most ρ doorways.
  2. Select each section with probability $\frac{1}{\ell}$, and choose any guard that covers it.

In all of these strategies, $V_m = \frac{1}{\ell}$. **P1** cannot decrease this since there are no doorways that are not covered with a probability less than $\frac{1}{\ell}$. **P2** cannot increase this since there are no two clusters that can be covered (even partially) by the same guard. Therefore, for each possible guard location, there is no set of doorways that the guard can see that will be selected by **P1** with a total probability greater than $\frac{1}{\ell}$.

6.1.2 Multiple guards

We now consider a corridor with n doorways, covered by m guards, each of which can guard ρ doorways. Section 6.1.1 is a special case of this, where $m = 1$. In this section we use domination to show that guards and intruder should be placed at positions indicated by the equilibrium strategies from Section 6.1.1.

As in previous examples, **P1** chooses his action $i \in \{1, \ldots, n\}$. In this example, **P2** chooses $J \in \left\{J_1, J_2, \ldots, J_{\binom{n}{m}}\right\}$, where each $J_\alpha = \{j_{\alpha 1}, \ldots, j_{\alpha m}\}$, and each $j \in \{1, \ldots, n - \rho + 1\}$ represents a guard at j that sees every $i \in \{j, \ldots, j + \rho - 1]$. Without loss of generality, $j_{\alpha 1} < j_{\alpha 2} < \cdots < j_{\alpha m}$.

We define these functions:

$$\begin{aligned}
\hat{\imath}\,(J) &= \{i \,|\, \exists j \in J \,\text{s.t.}\, i \in [j, j + \rho - 1]\,\} \\
\hat{J}\,(i) &= \{J \,|\, \exists j \in J \,\text{s.t.}\, i \in [j, j + \rho - 1]\,\}\,.
\end{aligned}$$

In other words, $\hat{i}\,(J)$ contains all the doorways that J covers, and $\hat{J}\,(i)$ contains all the guard deployments that cover i. Since the game outcome is always 1 or 0, we can thus define domination using $\hat{i}$ and $\hat{J}$:

- i_1 dominates i_2 iff $\hat{J}\,(i_1) \subset \hat{J}\,(i_2)$.
- J_1 dominates J_2 iff $\hat{i}\,(J_1) \supset \hat{i}\,(J_2)$.

(These relations are strict subset and superset.)

Using this, we follow a domination process that parallels the one used in Section 6.1.1.

Lemma 15: $i = 1$ dominates every $i \in \{2, \ldots, \rho\}$.

Proof: A deployment J that covers $i = 1$ must contain $j = 1$, so $\hat{J}\,(1) = \{J\,|1 \in J\}$. Such a deployment would also cover any $i \in \{2, \ldots, \rho\}$; therefore $\hat{J}\,(1) \subseteq \hat{J}\,(i)$. Since $\hat{J}\,(i)$ contains other deployments, of which $\{i, i+\rho, \ldots\}$ is one example, $\hat{J}\,(1) \subset \hat{J}\,(i)$. Therefore, $i = 1$ dominates $i \in \{2, \ldots, \rho\}$. $\square$

Lemma 16: $i = n$ dominates every $i \in \{n-\rho+1, \ldots, n-1\}$.

Proof: A deployment J that covers $i = n$ must contain $j = n-\rho+1$, so $\hat{J}\,(n) = \{J\,|(n-\rho+1) \in J\}$. Such a deployment would also cover any $i \in \{n-\rho+1, \ldots, n-1\}$; therefore $\hat{J}\,(n) \subseteq \hat{J}\,(i)$. Since $\hat{J}\,(i)$ contains other deployments, of which $\{i-\rho+1, i-2\rho+1, \ldots\}$ is one example, $\hat{J}\,(n) \subset \hat{J}\,(i)$. Therefore, $i = n$ dominates $i \in \{n-\rho+1, \ldots, n-1\}$. $\square$

Compare these two lemmas to lemmas 7 and 8. The next two lemmas demonstrate domination for columns, which represent guard deployment. While there are certain parallels to lemmas 12 and 13, these lemmas are very different, as they apply to multi-guard deployments.

Lemma 17: For any deployment J that contains a $j \in \{2, \ldots, \rho\}$, there is J' that dominates it.

Proof: Set $J' = (J - \{j\}) \cup \{\rho\}$. The replacement of j with ρ covers additional doorways between $j + \rho$ and $2\rho - 1$, in exchange for no longer covering the doorways between j and $\rho - 1$. But every $i \in \{j, \ldots, \rho - 1\}$ is dominated. Therefore $\hat{\imath}\,(J) \subseteq \hat{\imath}\,(J')$. The two sets are equal if some other guard in J covers these new doorways, i.e. J contains a $j' \in \{j + 1, \ldots, \rho\}$. If this is so, J' contains overlapping guards; we already know j' and ρ overlap, and there may be others. Therefore, we move the overlapping guards to the right until they no longer overlap. This adds elements to $\hat{\imath}\,(J')$ without losing any elements. Let $\{j_1, \ldots, j_p\} \subset J$ be the maximal set of consecutive overlapping or touching guards starting with j, i.e. $j_1 = j$ and $\forall q \in \{1, \ldots, p\}, j_q + \rho \geq j_{q+1}$. Set $J' = (J - \{j_1, \ldots, j_p\}) \cup \{\rho, 2\rho, \ldots, p\rho\}$. In either case, $\hat{\imath}\,(J) \subset \hat{\imath}\,(J')$, so J' dominates J. Therefore, any guard deployment that contains a $j \in \{2, \ldots, \rho\}$ is dominated. $\square$

Lemma 18: For any deployment J that contains a $j \in \{n - 2\rho + 2, \ldots, n - \rho\}$, there is J' that dominates it.

Proof: Set $J' = (J - \{j\}) \cup \{n - 2\rho + 1\}$. The replacement of j with $n - 2\rho + 1$ covers additional doorways between $n - 2\rho + 1$ and $j - 1$, in exchange for no longer covering the doorways between $n - \rho + 1$ and $j + \rho - 1$. But every $i \in \{n - \rho + 1, \ldots, j + \rho - 1\}$ is dominated. Therefore $\hat{\imath}\,(J) \subseteq \hat{\imath}\,(J')$. The two sets are equal if some other guard in J covers these new doorways, i.e. J contains a $j' \in \{n - 2\rho + 1, \ldots, j - 1\}$. If this is so, J' contains overlapping guards; we already know j' and $n-2\rho+1$ overlap, and there may be others. Therefore, we move the overlapping guards to the left until they no longer overlap. This adds elements to $\hat{\imath}\,(J')$ without losing any elements. Let $\{j_1, \ldots, j_p\} \subset J$ be the maximal set of consecutive overlapping or touching guards ending with j, i.e. $j_p = j$ and $\forall q \in \{1, \ldots, p\}, j_q +$

$\rho \geq j_{q+1}$. Set $J' = (J - \{j_1, \ldots, j_p\}) \cup \{n-(p+1)\,\rho+1, n-p\rho+1-j_p, \ldots, n-2\rho+1\}$. In either case (whether $\hat{\imath}\,(J) = \hat{\imath}\,(J')$ or not), $\hat{\imath}\,(J) \subset \hat{\imath}\,(J')$, so J' dominates J. Therefore, any deployment that contains a $j \in \{n-2\rho+2, \ldots, n-\rho\}$ is dominated. □

We continue this process, analogous to removing guards and intruder actions described in Section 6.1.1. In this game, instead of removing guards, we remove deployments containing guards at those locations. Like in Section 6.1.1, the process stops when looking at the overlapping guards in the middle. These guard locations are the locations represented by the rectangular block of 1s in the center of (6.6). Every guard in this area overlaps other guards in this area, but no guard dominates any other. Any attempt at new dominations will fail, as the J' constructed to dominate J has already been removed by this point. This process terminates with the same action set for **P1**; **P2**'s action set contains only sets of guards from the final action set in the one-guard game of Section 6.1.1. Therefore, if $\mathcal{I}$ and $\mathcal{J}$ are the action sets in the one-guard case for **P1** and **P2** respectively, then $\mathcal{I}$ and $\{J\,|\,J \subset \mathcal{J}, |J| = m\}$ are the action sets in the multi-guard case. As in Section 6.1.1, these strategies are not unique.

We apply strategies from Sections 5.2.1 and 6.1.1 to this game. Set $\ell = \left\lceil \frac{n}{\rho} \right\rceil$. The equilibrium strategies are:

- For **P1**:

  1. Select ℓ clusters of doorways, such that no two doorways from different clusters are less than ρ doors apart.

  2. Select each cluster with probability $\frac{1}{\ell}$, and choose any doorway in the cluster to traverse.

- For **P2**:

1. Divide the line of doorways into ℓ contiguous sections, each of which is at most ρ doorways wide.

2. Select m sections, so that each section is selected with probability $\frac{m}{\ell}$, and choose any set of guards that cover these m sections.

Notice that the intruder's strategy depends on n and ρ, but not m.

6.2 One continuous corridor

Consider a single vertical continuous corridor of width w, where there are m guards with range r. We apply the discrete-to-continuous conversion from Section 5.2.2. Select an arbitrary horizontal line $\mathcal{L}$ and $n \in \mathbb{N}$ as the number of equal sections in which to divide $\mathcal{L}$. The discrete examples can approximate the continuous case by setting $\rho = \arg\min_{\rho \in \mathbb{Z}^+} \left|\frac{\rho}{n} - \frac{r}{w}\right|$. In other words, select ρ so that $\frac{\rho}{n}$ best approximates $\frac{r}{w}$. Use the strategy from Section 6.1.2 to determine where **P2** will place guards and which points in $\mathcal{L}$ **P1** will traverse in intrusion paths. Once **P1** has selected point $x \in \mathcal{L}$, the intruder will traverse the vertical line from $\mathcal{S}_1$ to $\mathcal{S}_2$ through x. The value of this discrete game is $\dfrac{m}{\left\lceil \frac{n}{\rho} \right\rceil}$. As n increases, $\frac{\rho}{n}$ approaches $\frac{r}{w}$; for large enough n, $\left\lceil \frac{n}{\rho} \right\rceil = \left\lceil \frac{w}{r} \right\rceil$. Therefore the values of the continuous game is $\dfrac{m}{\left\lceil \frac{w}{r} \right\rceil}$. The equilibrium strategies are the same as in Section 6.1.2, only with $r, w \in \mathbb{R}^+$, and $\ell = \left\lceil \frac{w}{r} \right\rceil$.

One type of equilibrium strategy for **P1** here involves selecting ℓ vertical paths that are separated by at least distance r, and using each one with probability $\frac{1}{\ell}$. When **P1** uses this strategy, no guard can cover multiple candidate paths at once. Thus the game's outcome is at most $\frac{m}{\ell}$. In the equilibrium strategies given above, **P2** will cover each segment in the minimum fixed-length barrier with probability $\frac{m}{\ell}$. Since the intruder must cross one of these segments, the expected outcome is

$\frac{m}{\ell}$. Therefore the game's value is $V_m = \frac{m}{\ell}$. With this in mind, we can construct equilibrium strategies for any one-corridor environment, vertical or otherwise.

Definition: Paths $\gamma_1, \ldots, \gamma_n$ are *d-distant* if $\forall t_1, t_2 \in [0,1], \forall i, j \in \{1, \ldots, n\}$, where $i \neq j$,

$\|\gamma_i (t_2) - \gamma_j (t_2)\| \geq r$. In other words, if paths are d-distant, then they are always a distance of at least r apart from each other[2].

We call a workspace $\mathcal{W}$ a *corridor* if it is simply connected, and $\mathcal{S}_1$ and $\mathcal{S}_2$ are each one component, and contained in the boundary of $\mathcal{W}$. For a corridor, all intrusion paths are homotopic to each other, and the minimum variable-length barrier is a single segment $\mathcal{L}$ of length w [13]. Let $M = \left\lceil \frac{w}{r} \right\rceil$ be the size of the minimum fixed-length barrier, produced by placing guards across the minimum variable-length barrier. **P2**'s equilibrium strategy is to place guards across $\mathcal{L}$ in the same manner as the strategy in Section 6.1.2. **P1**'s strategy is to find M d-distant intrusion paths, and select each one with equal probability. Each intrusion path crosses at least one segment in the minimum fixed-length barrier, and each segment is covered by a guard with probability $\frac{m}{M}$. Similarly, each guard can detect at most one intruder. Therefore the coverage value is $\frac{m}{M}$.

It must be shown that it is always possible to construct M d-distant intrusion paths through $\mathcal{W}$.

Lemma 19: In a single-corridor environment, the maximum number of d-distant intrusion paths is equal to the size of the minimum fixed-length complete barrier of guards with length d.

Proof: Let ℓ be the maximum number of d-distant intrusion paths. Every such path must cross the minimum barrier; since the paths are d-distant, no two of them cross the same guard. Therefore $\ell \leq M$.

[2] In most cases we want the inequality to be strict. However the algorithms used here generally create paths that are exactly r apart. Therefore we use $\geq$ instead of $>$. See Section 6.4.2.

To show $\ell \geq M$, construct a set of M d-distant paths. This method parallels the decomposition method in Section 3.2.2. Since $\mathcal{W}$ is a corridor, $\mathcal{S}_1$ and $\mathcal{S}_2$ are on its boundary. Therefore they divide $\partial\mathcal{W}$ into two components, T_1 and T_2. Set $\gamma_1 = T_1$, $R_1 = V_{circ}(\gamma_1)$, and for each $i = 2, \ldots, M$, set $\gamma_i = \partial R_{i-1} - \partial W - \bigcup_{j=1}^{i-2} \overline{R}_j$ and $R_i = V_{circ}(\gamma_i) - \bigcup_{j=1}^{i-2} R_j$. If there is some $M' \leq M$ such that $\gamma_{M'}$ is broken into multiple components, then it intersects T_2 at some point p. If so, there is a chain of $M' - 1 < M$ guards from T_1 to $p \in T_2$. This is a barrier with a length less than M, which is a contradiction. Therefore, all γ_i paths are unbroken, and there are M such paths. $\square$

Distant paths resemble *thick non-crossing paths* [42]. A thick path of thickness δ (i.e. a δ-thick path) is a path through $\mathcal{W}$, plus all points within distance δ of the path. The thick-paths problem is that of finding the set of paths of minimum total length such that (1) the paths connect given endpoints, (2) no thick paths intersect, and (3) all thick paths are contained inside $\mathcal{W}$, i.e. the thin paths never get within a distance of δ from obstacles. The minimum thick non-crossing paths for a particular simply-connected domain can be found in linear time using an algorithm given in [42]. The only difference between $\frac{d}{2}$-thick paths and d-distant paths is the requirement that thick paths never overlap the boundary. Therefore, this algorithm can be applied to quickly find minimum d-distant paths for the intruder simply by removing the step where all points within distance δ from the boundary are removed from consideration in thick paths.

Notice that for single corridors, the strategies ignore whether the guards are segments or circles. Here r is the maximum distance across a guard's visibility region, which is the length of a segment and the diameter of a circle.

6.3 General polygonal environments

In this section we apply the concepts from the previous section to partial coverage strategies in general environments. We show how to combine full coverage, d-distant paths, and mixed game theory strategies to produce equilibrium strategies for both players. In Section 6.3.1, we show how d-distant paths apply directly to intruder strategies. In Section 6.3.2, we show that for segment guards, the strategies do not come directly from d-distant paths, but can be determined by manipulating the strategies from Section 6.3.1.

6.3.1 Circles

Consider an arbitrary polygonal environment $\mathcal{W}$, where $\mathcal{S}_1$ and $\mathcal{S}_2$ can be separated by a minimum of M guards with radius $r = \frac{d}{2}$. If there are $m < M$ available guards, **P2** can guarantee an outcome of at least $\frac{m}{M}$ by placing guards so that each guard in the complete barrier is covered with equal probability.

Now we show that **P1** can guarantee an outcome of at most $\frac{m}{M}$ by using d-distant paths. If **P1** selects ℓ d-distant paths, then each guard can only cover at most one intrusion path. Therefore, if **P1** selects each path with equal probability, then he will be detected with a probability of $\frac{m}{\ell}$. It remains to show that $\ell = M$, even in the presence of holes and multiple $\mathcal{S}_i$ components.

Theorem 5: In general polygonal environments, the maximum number of d-distant paths is equal to the size of the minimum fixed-radius complete circle barrier of guard with diameter d.

Proof: This theorem, and the proof, parallel the max-flow/min-cut duality [23]. Here, a cut is a complete barrier, and a flow is a set of d-distant paths. We will use augmenting paths to show that max flow=min cut in this environment, i.e. maximum number of d-distant paths = minimum complete barrier.

Let ℓ be the maximum number of d-distant intrusion paths, and M be the number of guards in the minimum circle barrier. As with Lemma 19, every d-distant intrusion path must cross the minimum barrier, and since the paths are d-distant, every path must cross a different guard. Therefore $\ell \leq M$.

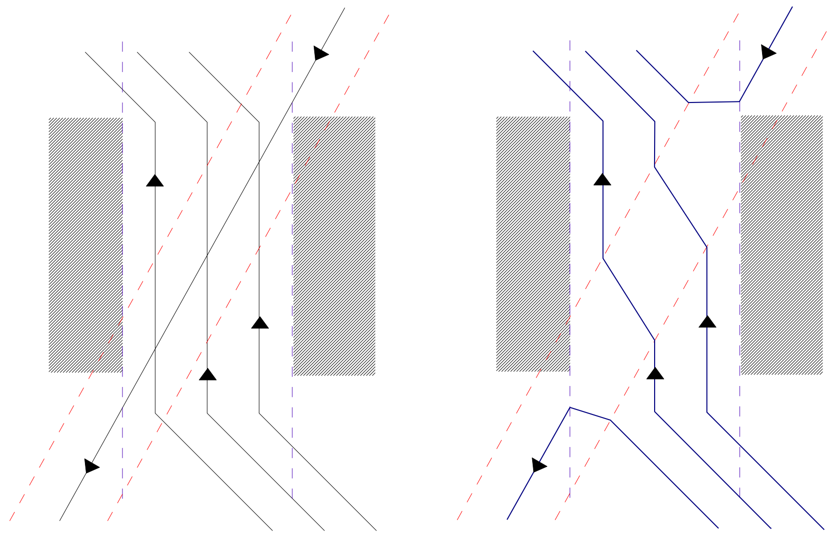
To show $\ell \geq M$, consider a set of d-distant paths $\gamma_1, \ldots, \gamma_{M'}$, where $M' < M$. We will show that one more intrusion path γ^* can be added. This γ^* is an augmenting path. It only moves through portions of $\mathcal{W}$ where either (1) there are no paths within distance d, (2) the paths within distance d are in the opposite direction, or (3) the paths move through corridors where there is room for one more path. Once γ^* is found, d-distant paths $\gamma'_1, \ldots, \gamma'_{M'+1}$ can be constructed.

To show γ^* exists, consider all points reachable from $\mathcal{S}_1$ while remaining d-distant from the γ_i paths. If at least one such point is in $\mathcal{S}_2$, then finding a γ^* to connect $\mathcal{S}_1$ to this point is straightforward. If not, consider every corridor containing γ_i paths that prevent the insertion of a γ^*.

If every intrusion path γ^* goes through a corridor where the number of γ_i paths is maximal and the paths move in the same direction as γ^*, then one can construct a minimum barrier by taking the union of the minimum barrier of each of the fully-utilized corridors. There will be one guard per γ_i path, otherwise some paths would be oriented in the opposite direction (i.e. moving away from $\mathcal{S}_2$ and towards $\mathcal{S}_1$). This is a minimum barrier of $M' < M$, which contradicts that the minimum barrier is of size M. Therefore, there is a γ^* that crosses only corridors where the γ_i paths move in the opposite direction, or there is room for at least one more d-distant path.

Once γ^* is found, it is used to construct $\gamma'_1, \ldots, \gamma'_{M'+1}$. If γ^* is already a distance of d away from all other paths, then $\gamma'_i = \gamma_i$ for $i = 1, \ldots, M'$, and $\gamma'_{M'-1} = \gamma^*$.

If γ^* traverses a corridor in the opposite direction as the paths $\gamma_{i_1}, \ldots, \gamma_{i_g}$, ordered in the same order γ^* approaches them, then the γ'_i paths can be found by rearranging pieces of these paths. For each j, let p_j and q_j be the points where γ_{i_j}

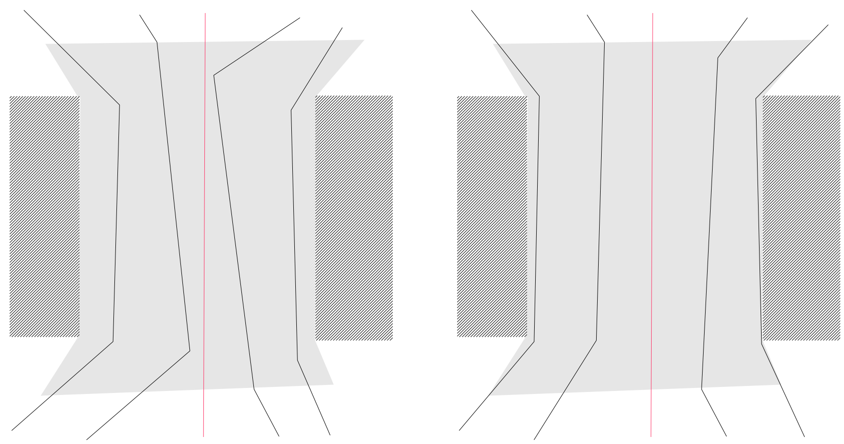


(a) Before (b) After

Figure 6.2: Augmenting path moving opposite directions as distant paths. Dashed lines are relevant boundaries of thick paths.

respectively enters and leaves the r-thick path about γ^*. Similarly, let p^* be the point where γ^* enters the r-thick path about γ_{i_1}, and let q^* be the point where γ^* exits the r-thick path about γ_{i_g}. Then for $j = 1, \ldots, g-1$, γ'_{i_j} is the portion of γ_{i_j} from $\mathcal{S}_1$ to x_j, followed by the line segment from p_j to q_{j+1}, followed by the portion of $\gamma_{i_{j+1}}$ from q_{j+1} to $\mathcal{S}_2$. $\gamma'_{M'+1}$ is the portion of γ^* from $\mathcal{S}_1$ to p^*, followed by the line from p^* to q_1, followed by the portion of γ_{i_1} from q_1 to $\mathcal{S}_2$. Similarly, γ'_g is the portion of γ_{i_g} from $\mathcal{S}_1$ to p_g, followed by the line segment from p_g to q^*, followed by the portion of γ^* from q^* to $\mathcal{S}_2$. For all other i, $\gamma'_i = \gamma_i$. See Figure 6.2 for an example. This step is analogous to reducing flow through an edge e via an augmenting path in the opposite direction of the flow through e.

If γ^* traverses a corridor of width greater than kd along with paths $\gamma_{i_1}, \ldots, \gamma_{i_k}$, the paths can be adjusted to make room for γ^*. Construct a corridor starting at the point where γ^* enters the d-thick path of some γ_{i_j} and ending at the point where γ^* leaves the d-thick path of some $\gamma_{i_{j'}}$. This corridor has thickness greater than k, so all $k+1$ paths can be d-distant, and connect with the paths outside the corridor. Each



(a) Before (b) After

Figure 6.3: Augmenting path adding to distant paths. The light grey area is the corridor in which the new path is found.

γ'_{i_j} consists of the portions of γ_{i_j} outside the corridor, plus the new d-distant path inside the corridor. $\gamma'_{M'+1}$ is defined analogously from γ^*. For all other i, $\gamma'_i = \gamma_i$. See Figure 6.3 for an example. This step is analogous to increasing flow through an edge e via an augmenting path in the direction of the flow through e.

Both of these steps can be applied multiple times if relevant. For example, if γ^* crosses two different corridors in the opposite direction as the γ_i paths. After applying them enough times, the result is $M' + 1$ d-distant paths. Therefore, any set of fewer than M d-distant paths can be added to, so the maximum number of d-distant paths must be M. $\square$

Since the number of d-distant paths is the same as the number of guards in the minimum circle barrier, both players can use this to construct equilibrium strategies. **P1** finds M d-distant intrusion paths, and selects each one with probability $\frac{1}{M}$. **P2** selects a minimum circle barrier, and covers each guard location with probability $\frac{m}{M}$. The resulting value is $V_m = \frac{m}{M}$.

6.3.2 Segments

In this section, we look at the problem of partial barrier coverage where there are m segment guards of length r. We will take the results from Section 6.3.1, and alter them to suit segment guards.

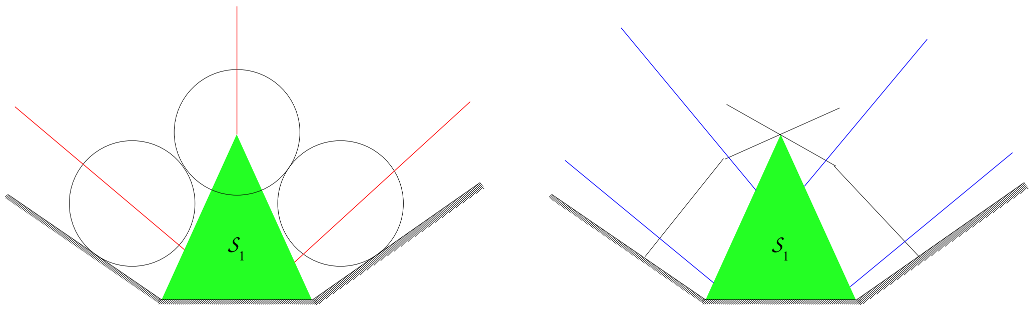
P2's strategy is the same for segments as for circles. **P2** finds a minimum fixed-length segment barrier, and places each of his m guards so that each segment in the minimum barrier is covered with a probability $\frac{m}{M}$, where M is the number of guards in the minimum barrier. Since the intruder has to cross one of these segments, the outcome is at least $\frac{m}{M}$.

In the previous section, we saw that there is an equivalence between circle barriers and distant paths. This is in part because two paths can be covered by the same guard diameter d iff the distance between them is less than d. This is not true of segment guards. Thus **P1**'s strategy with segment guards is different from the strategy in Section 6.3.1. We will see examples where paths that are within r cannot be covered by a single guard. Such situations happen when the minimum radius-r circle barrier contains fewer guards than the minimum length-$2r$ segment barrier.

Section 4.3.1 gives the three cases that lead to the segment barrier being larger than the circle barrier. For each case, we show how to select appropriate intruder paths.

For polygonal curve guard chains like in Figure 4.7, **P1** selects paths such that any two paths that are within r of each other can only be connected by line segments through $\mathcal{S}_i$. Figure 6.4 shows an example of this.

If the minimum circle barrier contains a guard covering an entire $\mathcal{S}_i$ component, which would require multiple segment guards, **P1** surrounds the $\mathcal{S}_i$ component with the minimum number of necessary segment guards, and chooses a path through each guard with equal probability.



(a) Circles (b) Segments

Figure 6.4: Guards and corresponding intrusion paths.

In circle barriers where one guard touches multiple obstacles or guards, **P1** constructs twice as many intruder paths, and traverses each one with probability $\frac{1}{2M}$, i.e. half as often as a regular path. Figure 6.5 shows the candidate paths for the example workspace from Figure 4.8(b). To guarantee each segment guard is traversed with equal probability $\frac{1}{2}$, each segment must be traversed with probability $\frac{1}{4}$.

This gives **P1** a set of paths he can select so that each path selected with probability $\frac{1}{M}$ can only be detected by one guard, and each path selected with probability $\frac{1}{2M}$ has at most one other such path that crosses the same guard. Since each such guard is covered with probability $\frac{m}{M}$, the outcome is at most $\frac{m}{M}$. Therefore, the game's value is $\frac{m}{M}$.

6.4 Additional notes

6.4.1 Intractability

For both types of guards and both players, finding the optimal strategy is NP-Complete. The problem of finding the minimum barrier is NP-Complete, as proven in [13] (for segments) and Section 4.1 (for circles). The problem of finding maximum d-distant paths is NP-Complete, as proven in [42]. Therefore, both players are likely

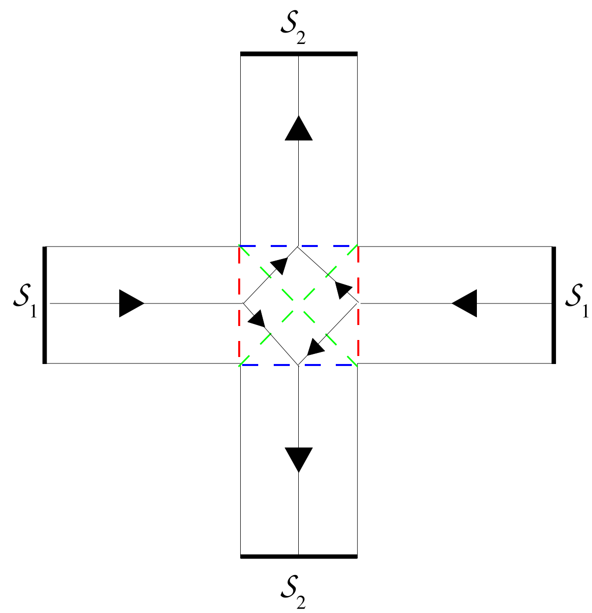


Figure 6.5: Intruder paths for environment in Figure 4.8. Any dashed pair of the same color is a segment barrier.

to use approximate methods to give strategies that approach equilibrium. **P1** may use fewer paths (raising the outcome), and **P2** may choose more locations in which to deploy guards (decreasing the outcome).

6.4.2 Closed guards

The sensor model used throughout this work assumes a guard with range r can see points at a distance up and including r. Therefore, if two paths are a distance r apart, a perfectly placed guard can cover both of them. Therefore, equilibrium intruder strategies require paths that are more than r apart.

However, the algorithms given are designed to construct paths that are exactly r apart. Therefore, in these cases it is preferable to use $r+\epsilon$-distant paths, for a small enough ϵ that adding it to r does not change the number of paths. For example, if the minimum barrier consists of chains of lengths $w_1, \ldots, w_n$, each chain i will contain $m_i = \left\lceil \frac{w_i}{r} \right\rceil$ guards. Therefore, if $\epsilon < \min_i \frac{w_i - r m_i}{m_i}$, then guards of length $r + \epsilon$

will produce the same m_i values. Therefore $r + \epsilon$ is a suitable distance to separate intruder paths.

Chapter 7

Future research

This chapter describes some other directions that can be taken in barrier coverage.

7.1 General intruders and guards

We can extend the barrier coverage problem definition in the case where the intruder is not a point, but a moving object with some configuration space $\mathcal{Q}$ and free space $\mathcal{Q}_{free} \subset \mathcal{Q}$, possibly with motion constraints. The sets $\mathcal{S}_1$, $\mathcal{S}_2$, and $V(q_j)$ retain their meaning, only now they are subsets of $\mathcal{Q}_{free}$. This reflects that in this case we are concerned with start, goal, and visibility *configurations*, not just locations.

In many cases, $\mathcal{S}_1$, $\mathcal{S}_2$, and $V(q_j)$ are effectively sets of locations. For example, guards see intruders in specific regions of space, regardless of how they are oriented. In these cases, it is worthwhile to define $\tilde{\mathcal{S}}_1$, $\tilde{\mathcal{S}}_2$, and $\tilde{V}(q_j)$ as, respectively, the set of start locations, stop locations, and locations seen by guards. If the intruder has the volume function $\mathcal{R} : \mathcal{Q} \to \mathbb{R}^m$, (A robot at $q \in \mathcal{Q}$ resides in $\mathcal{R}(q) \subset \mathbb{R}^m$), then we can write:

$$\begin{aligned} \mathcal{S}_1 &= \left\{ q \in \mathcal{Q}_{free} \middle| \mathcal{R}(q) \cap \tilde{S}_1 \neq \emptyset \right\} \\ \mathcal{S}_2 &= \left\{ q \in \mathcal{Q}_{free} \middle| \mathcal{R}(q) \cap \tilde{S}_2 \neq \emptyset \right\} \\ V(q_i) &= \left\{ q \in \mathcal{Q}_{free} \middle| \mathcal{R}(q) \cap \tilde{V}(q_i) \neq \emptyset \right\} \end{aligned}$$

We can also invert these definitions, e.g. define $\tilde{\mathcal{S}}_1$ in terms of $\mathcal{S}_1$.

This general definition can be used to represent three dimensions, multiple intruders, intruders with motion constraints (e.g. nonholonomic intruders, dynamic constraints, etc.), intruders with volume, intruders that change shape, etc.

We can extend the definition of guards the same way. Each guard q_j has a configuration space Q_j. Each guard can have a different configuration space, representing a set of guards with different properties, for example, guards that have to be on the ground, along walls or ceilings, or on specific tracks (for motion or power purposes).

Each Q_j would have its own $V_j : Q_j \to \mathcal{P}(\mathcal{Q})$, where $\mathcal{P}(\mathcal{Q})$ is the power set of $\mathcal{Q}$. This can be used to define various types of guards, like lines, segments, circles, cylinders, spheres, etc.

This definition permits guards in three dimensions. A single line is of limited use in three dimensions. But a set of lines close enough together can function like a plane if the intruder has volume. This could also be generated by one quick-moving line. A sector or a cone could be produced similarly.

7.2 Noisy sensors

In previous chapters we have assumed that a sensor will always detect an intruder that enters the sensor's range. That is not always a realistic assumption. There is usually a level of noise in the sensing, so there is some nonzero probability of a false negative. This can be modeled simply by a probability of failure, or more elaborately as a probability function, where the location relative to the sensor affects the sensor's accuracy. In either case, one can replace the visibility functions V_{seg}, V_{circ}, etc. used throughout the previous chapters with probability functions $PV_i : Q_i \times \mathcal{W} \to [0, 1]$, where $PV_i(q_i, p)$ is the probability that a guard with configuration q_i can see an intruder at p. For example, the further away the intruder is from the sensor, the less likely the sensor is to detect him. Also, the environment may be a factor: there may be regions that are better lit, dustier, etc., and this may affect sensor accuracy.

7.2.1 Exposure

In a common model for sensors, the probability of a guard detecting an intruder is a function of the intruder's location relative to the guards, and the time the intruder spends in range. One way of modeling this is with a *Sensor Field Intensity* value [35]. For a single intruder location, each sensor has an intensity value, e.g.

$$S(s,p) = \frac{\lambda}{[d(s,p)]^K}$$

where $d(s,p)$ is the Euclidean distance between the sensor s and the point p, and positive constants λ and K are system parameters. This represents the simple example where the further away an intruder is, the less likely he is to be detected. The sensor model may incorporate additional environmental factors.

The entire field has an intensity value $I(p)$ defined by combining the individual sensor intensities. This could be a weighted sum of all sensor values, or the highest value. The *exposure* over a single intruder path $\gamma : [0,1] \to \mathcal{W}$ is then

$$E(\gamma) = \int_0^1 I(\gamma(t)) \left| \frac{d\gamma}{dt} \right| dt.$$

This value can be converted to a probability.

$$p(\gamma) = 1 - e^{-E(\gamma)}$$

This $p(\gamma)$ can be used as the value of the intruder/guard game. As before, the intruder seeks to minimize this, while the guard deployer seeks to maximize it.

A relevant concept is the *minimum exposure path* [35, 52]. This is the path that minimizes E for a given sensor deployment. For the distance-only model described above, this path traverses edges of the Voronoi diagram of the sensor locations. An intruder that knows where the guards are will traverse this path. An intruder that does not know the guard locations will seek the path of minimum *expected* exposure. This will produce game strategies like in the deterministic sensor case.

7.2.2 k-coverage

The concept of k-coverage from [31] may be useful here. A passage is k-covered if it is impossible for an intruder to travel from one end of the passage to the other without passing through k sensors' visibility regions. If the probability of a single failure is p, the probability of crossing the passage completely unseen is p^k. If the probability depends on location relative to the guard, then the guards could be staggered so that one guard sees well where another sees poorly.

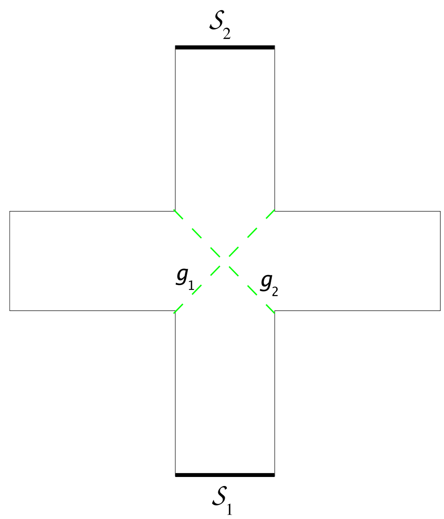
We define a k-barrier to be a guard deployment that k-covers every intrusion path. We show that for segment barriers (variable or fixed length), a minimum k-barrier can be found by deploying k copies of the minimum 1-barrier. We then show the same for circles.

Lemma 20: A segment k-barrier where no two guards' interiors intersect is composed of k disjoint 1-barriers.

Proof: We prove this inductively by demonstrating that every k-barrier B can be separated into a 1-barrier B_1 and a $(k-1)$-barrier B_{k-1} where the two barriers are disjoint. This is trivially true for $k = 1$.

For each intrusion path, consider the first guard that the path crosses. The 1-barrier B_1 is the set of all such guards for all intrusion paths. This is a 1-barrier since every intrusion path must cross at least one guard.

To show that $B_{k-1} = B - B_1$ is a $(k-1)$-barrier, it suffices to show that every intrusion path crosses at least $k-1$ guards in B_{k-1}. Prove by contradiction: suppose



(a) Guards intersect (b) Guards do not intersect

Figure 7.1: Equivalent segment guards

this is not the case. Then there exists a path α that crosses $k - i$ guards in B_{k-1}, where $i > 1$. Since B is a k-guard, α must cross at least k guards in B. Therefore, α crosses at least i guards in B_1. Let g^* be the last such guard. By construction there exists a path β which crosses g^* before any other guard in B. Express α as $\alpha_1 \cdot \alpha_2$, where α_1 connects $\mathcal{S}_1$ to g^*, and α_1 connects g^* to $\mathcal{S}_1$. Express β as $\beta_1 \cdot \beta_2$ analogously. The intrusion path $\beta_1 \cdot \alpha_2$ crosses $k - i + 1 < k$ guards, which contradicts the fact that B is a k-barrier. Therefore no such path exists, and B_{k-1} is a $(k-1)$-barrier.

Continuing this process decomposes B into k 1-barriers. $\square$

This method of extracting individual barriers also gives an order for guards. An intruder will go through the first extracted 1-barrier, then the second, etc. until the last. This concept breaks down when guards can intersect, like in Figure 7.1(a). There are paths in the same homotopy class that traverse g_1 and g_2 in opposite orders. In the proof below, we will use the fact that there are equivalent deployments like 7.1(b) that do not intersect.

Theorem 6: The minimum segment k-barrier consists of k copies of the minimum segment 1-barrier.

Proof: Let W be the length of the minimum segment barrier. Since according to Lemma 20 any non-intersecting segment k-barrier consists of k 1-barriers, its total length must be at least kW. This is achieved with n copies of the minimum 1-barrier.

Now consider a k-barrier where the guards do intersect. We know from Theorem 1 in Section 2.2 and the circle guard discussion in Section 4.3.1 that any set of intersecting guards can be replaced with non-intersecting guards of equal or lesser length, then any intersecting k-barrier is at least as long as some non-intersecting k-barrier, which is at least as long as k minimum 1-barriers.

Therefore, k minimum 1-barriers is a minimum k-barrier. □

Theorem 7: The minimum circle k-barrier consists of k copies of the minimum segment 1-barrier.

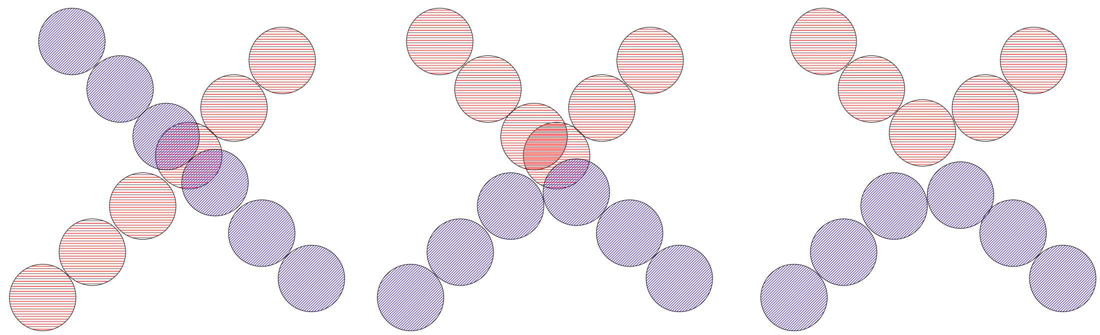
Proof: If a circle guard deployment, after being converted to its equivalent segment guard (see Section 4.3.1), contains no intersections, then proceed as in Theorem 6.

If the circles still intersect, there are three possibilities:

- Chains intersect (Figure 7.2). Use the same methods as in Theorems 1 and 6 to shorten and separate them.

- Individual circles overlap (Figure 7.3). In this case, either (1) intruders can always choose one circle to traverse and ignore the other (the vertical paths), or (2) intruders always both circles in the same order (the horizontal path). The only way to make this impossible is to use such a narrow corridor that the circles can be moved to coincide (see below). Therefore, there is no conflict like with segments.

- Circles coincide. In this case, merely assign an order that allows the algorithm



(a) Chains cross (b) Chains touch (c) Chains do not intersect

Figure 7.2: Intersecting circle guard chains can be rearranged to form nonintersecting chains.

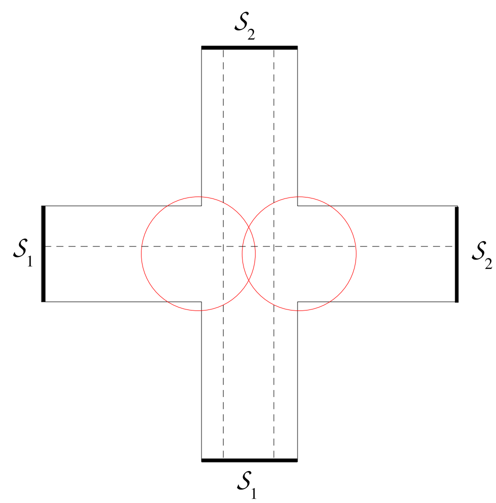
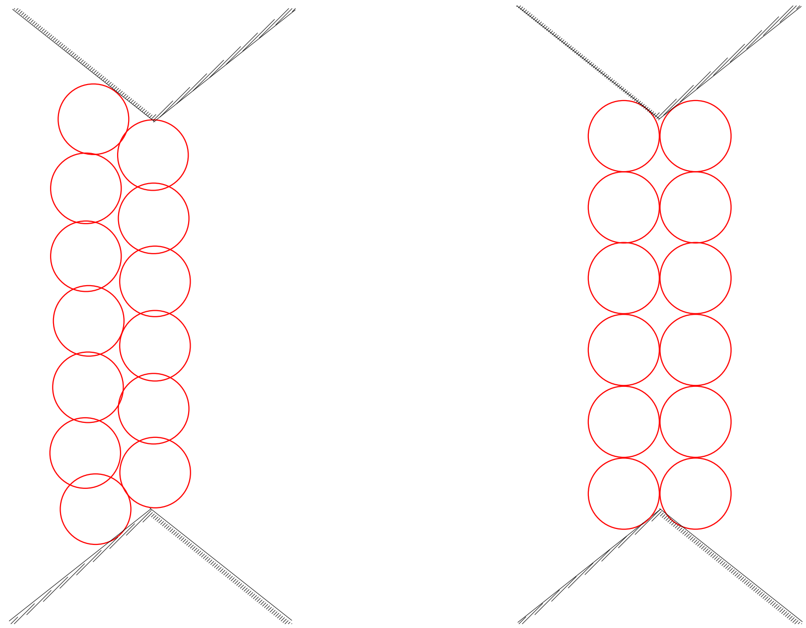


Figure 7.3: Overlapping circle guards.

in Lemma 20 to succeed: if g_1 and g_2 coincide, and γ crosses it n times, g_i is crossed first on odd-numbered crossings, and g_j is crossed first on the even numbered crossings. This puts g_i in the 1-barrier immediately before g_j.

Therefore, every circle k-barrier is at least as long as k copies of the minimum circle k-barrier. □

A likely practical constraint is that guards be a certain distance apart. In such cases, start with the minimum barrier of the given guard type, then use the algorithm in Section 3.2.3 to find the shortest barrier that does not touch the first barrier. It



(a) Minimum and second-minimum barriers (b) Minimum barrier pair

Figure 7.4: If guards must be a distance apart, the minimum 2-barrier is not always the minimum plus by the second-minimum.

will be necessary to make two modifications: (1) add edges corresponding to placing the second barrier next to the first, and (2) look for cases like Figure 7.4, where the minimum barrier pair does not contain the minimum barrier. Such cases require that the two barriers be adjacent.

7.2.3 Tracking

A guard may also have false positives, and it may be undesirable to have guards sound alarms, shut down the system, etc., if there is no intrusion attempt. In this case, it will be necessary for multiple nearby guards to see an intruder in order for the system to consider an intruder to be detected. This relates to the general problem of *tracking* in sensor networks [34]. This is the problem of determining the location and course of an object moving through a sensor field. In the deterministic sensor case, it is only necessary to detect the intruder once. If there is a possibility of a false positive, the sensors will need to communicate and work together to determine if they really are seeing an intruder. For example, the intruder should trip multiple sensors in a realistic order. Such a situation will also require a level of k-coverage. If there is only one guard protecting an area, there is no way of confirming or denying the sensor results.

7.3 Sleeping sensors

Sensors often have limited power resources. One way to conserve energy is to be active a portion of the time and sleep the rest of the time. If the guards all sleep with independent schedules, or randomly, then they can be treated the same as probabilistic sensors. In [30], k-coverage is applied to a network of sensors that sleep.

Alternatively, the guards can synchronize their sleep patterns so that a complete k-barrier is up at any time, for some k. This can be done using predefined clock settings, or by checking neighbors' sleep/wake settings. There are existing algorithms to schedule sleep that maintain area coverage [53], and that maintain connectivity [6]. Maintaining a barrier would involve elements of both: maintain a connected line between sides of a corridor, while keeping a high level of barrier coverage.

7.4 Guard motion

It is possible to use guards that can move while the intruder is moving through $\mathcal{W}$. In a complete barrier, guards do not move, as there is no advantage of doing so. However in partial coverage scenarios, guards may find it advantageous to move around to cover partially-covered barrier candidates, or to move from component to path component. Guard motion affects the intruder's path, but it can also affect the intruder's path's homotopy class if the motion is dependent on location (e.g. if it is based on terrain).

Mobile guards have speed bounds, and may have kinematic and dynamic constraints as well. All of these constraints may be dependent on the guard's location in the workspace. The intruder will also have constraints, but they may be different. For example, the intruder may be on the ground while the guards are on the ceiling. Similarly, the guards may be wheeled robots which move better on smooth terrain, while the intruder is a walking robot which moves better on rough terrain. Each guard may have its own set of constraints.

On the simplest level, the coverage value for moving guards is the average over time of the stationary guard coverage values. However, if the guards are moving fast enough, they can intercept an intruder who was just in the gap between them and lacks enough time and maneuverability to escape. In this case mobile guards are preferable to stationary guards.

7.5 Environment motion

Environment motion involves moving walls, doors, and other changes to the terrain over time. As with guard motion, this happens without any knowledge of the intruder's location, and can be scheduled or randomized. However, unlike guard motion, this happens with a pattern that is not strategic, and is known to the guards and the intruder.

7.6 Network connectivity

The guards need to communicate with each other in order to work together. If they are communicating over a wireless network (which will be necessary if they are moving, or outside), this will add constraints. Each guard is a node in the wireless network, and each will have a communications range. They may also have line-of-site requirements (for example, if they use infrared to communicate). It may be necessary to add guards that don't see anything, but can relay messages. This places a trade-off between a shorter barrier with far-apart components and a larger barrier with components closer together.

Just because two nodes are within communications range does not mean they can communicate. There is always the possibility of random communications failure, and sometimes the environment is a factor. This adds to the false negative probability: a guard that sees an intruder but cannot communicate this detection with other guards has the same effect as a guard that does not see an intruder. Therefore the guards should always be alert for network disconnection. The guards need to know what to do if the communication network divides into multiple components; they also need to minimize this possibility. An intruder may also choose to jam the network to get through, or jam the network in a different place to misdirect the guards. Now a robust network becomes an additional requirement, to balance against minimum cost.

7.7 Deployment motion planning

The work in previous chapters is concerned with the locations in which to deploy guards. It is not concerned with how to get the guards to their desired locations. While one could simply select a deployment and use a simple algorithm to implement it, there are other ways to combine selection with implementation.

- Deploy on the fly using incremental methods. For example, use gradient descent on the barrier's current properties, and have each robot plan motion separately.
- Once the target deployment is found, plan a search or sweep algorithm starting from the current deployment to minimize the chance that an invader has reached its goal before the deployment finishes.

7.8 Suboptimal deployments

Chapters 5 and 6 give solutions for optimal partial coverage in situations where the guards can choose where to deploy. However, in some situations, guards are deployed randomly (e.g. random walks, airdropped sensors, etc.). In this case, the coverage will not be optimal, and the intruder will not necessarily move with the same assumptions.

One can use the intruder paths from Section 5.3.2 as **P1**'s strategy, and determine the coverage value to be the percentage of these paths covered by the guard deployment. However, it may give an artificially high value. In situations like in Figure 7.5, the intruder has an undetected intrusion path. However, if the intruder were to follow the equilibrium strategy, they would always be detected. Situations like this produce a discrepancy between the value determined by the above method, and a meaningful coverage value. Therefore new intruder strategies must be devised.

Guard strategies are also relevant here, but in a different capacity than before. A randomized deployment strategy (based on a distribution over $\mathcal{W}$, rather than over the minimum barrier) would aim to maximize the expected coverage value. A post-deployment motion strategy could help randomized mobile guards to move towards a locally-optimal deployment. One example of such a post-deployment strategy is to follow the intruder paths, but redirected toward the minimum barrier. This would

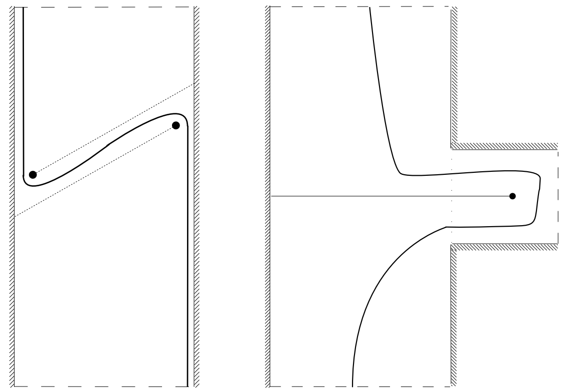


Figure 7.5: These intruder paths avoid the guards. However, all equilibrium intruder paths are vertical, and would be detected.

ultimately place all guards along the minimum barrier.

7.9 Multiple intrusion attempts

In the standard game for partial barriers, the intruder can make one intrusion attempt based on his current knowledge (or lack thereof) to intrude. If the intruder can make multiple attempts (each one ends upon successful guard detection), then the game becomes a learning process. Each player tries to learn the other's strategy.

In this scenario, the intruder **P1** and the guard deployer **P2** play a game where the outcome is the number of intrusion attempts used to successfully reach $\mathcal{S}_2$. **P1** seeks to minimize this, and **P2** seeks to maximize it.

Guard motion, briefly described in Section 7.4, affects the game. The speed constraints on the intruder can be relaxed, provided there is a time interval between attempts. This would mean the guards can reposition within bounds after each turn. Due to the turn-based nature of the game, there are three possibilities for motion constraints.

- Stationary guards

- Moving guards, bounded intruder speed
- Moving guards, unbounded intruder speed (with gaps between attempts)

References

[1] F. Aurenhammer. Voronoi diagrams – a survey of fundamental geometric data structure. *ACM Computing Surveys*, 23:345–405, September 991.

[2] T. Basar and G.J. Olsder. *Dynamic Noncooperative Game Theory*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.

[3] J. D. Boissonnat, J. Czyzowicz, O. Devillers, and M. Yvinec. Circular separability of polygons. *Algorithmica*, 30:67–82, 2001.

[4] Z. Butler, P. Corke, R. Peterson, and D. Rus. Virtual fences for controlling cows. In *Proc. IEEE ICRA '04. Int. Conf. Robotics and Automation 2004*, volume 5, pages 4429–4436, New Orleans, LA, April 2004.

[5] P. Chang. A closed-form solution for inverse kinematics of robot manipulators with redundancy. *IEEE Journal of Robotics and Automation*, 3:393–403, October 1987.

[6] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. *ACM/Kluwer Wireless Networks*, 8, 2002.

[7] Howie Choset. Coverage of known spaces: The boustrophedon cellular decomposition. *Autonomous Robots*, 9:247–253, 2000.

[8] Howie Choset. Coverage for robotics - a survey of recent results. *Annals of Mathematics and Artificial Intelligence*, 31:113–126, October 2001.

[9] V. Chvatal. A combinatorial theorem in plane geometry. *Journal of Combinatorial Theory Series B*, 18:39–41, 1975.

[10] V. Chvatal. A greedy heuristic for the set-covering problem. *Mathematics of operations research*, 4:233 –235, 1979.

[11] Thomas Clouqueur, Veradej Phipatanasuphorn, Parameswaran Ramanathan, and Kewal K. Saluja. Sensor deployment strategy for target detection. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 42–48, September 2002.

[12] R. Connelly, E.D. Demaine, and G. Rote. Straightening polygonal arcs and convexifying polygonal cycles. In *41st Annual Symposium on Foundations of Computer Science*, pages 432–442, 2000.

[13] Erik D. Demaine, Jeff Erickson, Ferran Hurtado, John Iaconom Stefan Langerman, Henk Meijer, Mark Overmars, and Sue Whitesides. Separating point sets in polygonal environments. In *Proceedings of the 20th Annual ACM Symposium on Computational Geometry*, 2004.

[14] H. Everett, G.A. Gilbreath, T.A. Heath-Pastore, and R.T. Laird. Coordinated control of multiple security robots. In *Mobile Robots VIII*, page 2058:292 305, 1993.

[15] Uriel Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

[16] L. R. Ford and D. R. Fulkerson. *Flows in Networks.* Princeton U. Press, Princeton, NJ, 1962.

[17] D. W. Gage. Command control for many-robot systems. In *AUVS92, the Nineteenth Annual AUVS Technical Symposium*, pages 22–24, Hunstville Alabama, USA, June 1992.

[18] Jacob E. Goodman and Joseph O'Rourke. *Handbook of Discrete and Computational Geometry.* Chapman & Hall, New York, 2004.

[19] A. Howard, M.J. Mataric, and G.S. Sukhatme. An incremental deployment algorithm for mobile robot teams. In *IEEE/RSJ International Conference on Intelligent Robots and System*, volume 3, pages 2849 – 2854, October 2002.

[20] Ferran Hurtado, Marc Noy, Pedro A. Ramos, and Carlos Seara. Separating objects in the plane by wedges and strips. *Discrete Applied Mathematics*, 109:109–138, 2001.

[21] S. Hynes and N. Rowe. Multiagent simulation for assessing massive sensor deployment. *Journal of Battlefield Technology*, 7(2):23–36, July 2004.

[22] David S. Johnson. Approximation algorithms for combinatorial problems. In *STOC '73: Proceedings of the fifth annual ACM symposium on Theory of computing*, pages 38–49, New York, NY, USA, 1973. ACM.

[23] L. R. Ford Jr. and D. R. Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404, 1956.

[24] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, New, New York, 1972.

[25] A.V. Karzanov. Determining the maximal flow in a network by the method of preflows. *Soviet Mathematics Doklady*, 15:434–437, 1974.

[26] V. Klee. Is every polygonal region illuminated from some point? *American Mathematics Monthly*, 76:180, 1969.

[27] C. Klein and L. Chu. Comparison of extended jacobian and lagrange multiplier based methods for resolving kinematic redundancy. *Journal of Intelligent and Robotic Systems*, 19:39–54, 1997.

[28] S. Kloder and S. Hutchinson. Barrier coverage for variable bounded-range line-of-sight guards. In *Proc. IEEE ICRA '07. Int. Conf. Robotics and Automation 2007*, volume 1, pages 391–396, Rome, Italy, April 2007.

[29] S. Kloder and S. Hutchinson. Partial barrier coverage: Using game theory to optimize probability of undetected intrusion in polygonal environments. In *Proc. IEEE ICRA '08. Int. Conf. Robotics and Automation 2007*, Pasadena, CA, May 2008.

[30] S. Kumar, T. H. Lai, and J. Balogh. On k-coverage in a mostly sleeping sensor network. In *International Conference on Mobile Computing and Networking (ACM MobiCom)*, pages 144–158, 2004.

[31] S. Kumar, T.H. Lai, and A. Arora. Barrier coverage with wireless sensors. In *Proceedings of the Eleventh Annual International Conference on Mobile Computing and Networking (ACM MobiCom)*, Cologne, Germany, 2005.

[32] D. Kurabayashi, J. Ota, T. Arai., and E. Yoshida. Cooperative sweeping by multiple mobile robots. In *Proc. IEEE International Conference on Robotics and Automation*, volume 2, pages 1744 – 1749, April 1996.

[33] L. Kučera. Finding a maximum flow in /s,t/-planar network in linear expected time. *Mathematical Foundations of Computer Science 1984*, 176:370–377, 1984.

[34] D. Li, K.D. Wong, Y.H. Hu, and A.M. Sayeed. Detection, classification, and tracking of targets. *IEEE Signal Processing Magazine*, 19:17–29, March 2002.

[35] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak. Exposure in wireless ad-hoc sensor networks. In *Proceedings of the 7th International Conference on Mobile Computing and Networking (MobiCom '01)*, pages 139–150, July 2001.

[36] Joseph S. B. Mitchell and Subhash Suri. Separation and approximation of polyhedral objects. *Computational Geometry*, 5:95–114, 1995.

[37] Ryan Morlok and Maria Gini. Dispersing robots in an unknown environment. In *DARS 2004*, Toulouse, France, June 2004.

[38] J. O'Rourke. *Art gallery theorems and algorithms*. Oxford University Press, New York, 1987.

[39] J. O'Rourke and K. Supowit. Some NP-hard polygon decomposition problems. *IEEE Transactions on Information Theory*, 29:181–190, March 1983.

[40] Praveen Paruchuri, Jonathan P. Pearce, Milind Tambe, Fernando Ordonez, and Sarit Kraus. An efficient heuristic approach for security against multiple adversaries. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2007)*, 2007.

[41] S. Poduri and G.S. Sukhatme. Constrained coverage for mobile sensor networks. In *Proc. IEEE ICRA '04. Int. Conf. Robotics and Automation 2004*, volume 1, pages 165 – 171, New Orleans, LA, April 2004.

[42] V. Polishchuk. *Thick Non-Crossing Paths and Minimum-Cost Continuous Flows in Geometric Domains*. PhD thesis, Stony Brook University, Stony Brook, NY, June 2007.

[43] Hans Rohnert. Shortest paths in the plane with convex polygonal obstacles. *Information Processing Letters*, 23(2):71–76, August 1986.

[44] M.-F.. Roy S. Basu, R. Pollack. On the combinatorial and algebraic complexity of quantifier elimination. *Journal of Associated Computer Machinery*, 43:1002–1045, 1996.

[45] M. Saitoh, Y. Takahashi, A. Sankaranarayanan, H. Ohmachi, and K. Marukawa. A mobile robot testbed with manipulator for security guard application. In *Proc. IEEE ICRA '95. Int. Conf. Robotics and Automation 1995*, volume 3, pages 2518–2523, May 1995.

[46] I. Streinu. A combinatorial approach to planar non-colliding robot arm motion planning. In *41st Annual Symposium on Foundations of Computer Science*, pages 443 – 453, November 2000.

[47] S. Suri. On some link distance problems in a simple polygon. *IEEE Transactions on Robotics and Automation*, 6:108–113, February 1990.

[48] A. Tarski. *A decision method for Elemntary Algebra and Geometry*. University of California Press, Berkeley, 1951.

[49] Godfried Toussaint. Movable separability of sets. In *Computational Geometry, 1985, North-Holland*. Elsevier Science Pub. Co, North Holland, 1985.

[50] B. Tovar, L. Guilamo, and S. M. LaValle. Gap navigation trees: Minimal representation for visibility-based tasks. In *Proc. Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2004.

[51] R. Vaughan, N. Sunipter, A. Frost, and S. Cameron. Robot sheepdog project achieves automatic flock control. In *Proc. Fifth International Conference on the Sirnulation of Adaptive Behavious*, pages 489–493, 1998.

[52] G. Veltri, Q. Huang, G. Qu, and M. Potkonjak. Minimal and maximal exposure path algorithms for wireless embedded sensor networks. In *ACM International Conf. on Embedded Networked Sensor Systems (SenSys)*, pages 40–50, 2003.

[53] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill. Integrated coverage and connectivity configuration in wireless sensor networks. In *ACM International Conf. on Embedded Networked Sensor Systems (SenSys)*, pages 28–39, 2003.

[54] D. West. *Introduction to Graph Theory.* Prentice Hall, Upper Saddle River, NJ, 2001.

[55] S.C. Wong and B.A. MacDonald. A topological coverage algorithm for mobile robots. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 1685–1690, October 2003.

[56] Alexander Zelinsky. A mobile robot exploration algorithm. *IEEE Transactions on Robotics and Automation*, 8:707–717, 1992.

[57] Xiaoming Zheng, Sonal Jain, Sven Koenig, and D. Kempe. Multi-robot forest coverage. In *Proc. IROS 2005. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005*, pages 3852–3857, August 2005.

Curriculum Vitae

Stephen Kloder

RESEARCH INTERESTS

Robot motion planning, learning, reasoning, classification, geometry

EDUCATION

Ph.D. in Computer Science, University of Illinois, 2008

Thesis: *Barrier Coverage: Deploying robot guards to prevent intrusion*

Advisor: Professor Seth Hutchinson

M.S. in Computer Science, University of Illinois, 2004

Thesis: *Permutation-invariant multi-robot formations: A Complex Polynomial Foundation*

Advisor: Professor Seth Hutchinson

B.S. (High Honors) in Computer Science with Cooperative Plan, Georgia Institute of Technology, 2001

PROFESSIONAL POSITIONS

Research Assistant Dept. of Electrical Engineering, University of Illinois, (2004-2006)

Research in multi-robot motion planning, and barrier coverage.

Teaching Assistantships

Term	Class	Responsibilities
Spring 2008	CS 433	Grading, writing assignments/tests
Spring,Fall 2007	CS 241	Discussion sessions, MP creation
Spring 2004	CS 101	Lab sessions
Fall 2003	CS 375	Grading, writing assignments/tests

AWARDS AND HONORARIES

- Finalist for Best Student Paper Award, "Partial barrier coverage: Using game theory to optimize probability of undetected intrusion in polygonal environments,", *IEEE International Conference on Robotics and Automation*, 2008
- Recipient, Conference Travel Grant award, Graduate College, UIUC, 2007

PRESENTATIONS

- *Partial barrier coverage: Using game theory to optimize probability of undetected intrusion in polygonal environments*, 4/22/08, Pasadena, CA
- *Barrier Coverage for Variable Bounded-Range Line-Of-Sight Guards*, 4/11/07, Rome, Italy
- *A Configuration space for permutation-invariant multi-robot formations*, 4/29/04, New Orleans, LA

Journal Articles

S. Kloder and S. Hutchinson, "Path planning for permutation-invariant multirobot formations," *IEEE Transactions on Robotics*, vol.22, no.4, pp. 650- 665, Aug. 2006.

Refereed Conference Papers

- S. Kloder and S. Hutchinson, "Partial barrier coverage: Using game theory to optimize probability of undetected intrusion in polygonal environments," *Proc. IEEE Int'l. Conf. on Robotics and Automation*, PAsadena, CA, 2008.

- S. Kloder and S. Hutchinson, "Barrier Coverage for Variable Bounded-Range Line-Of-Sight Guards," *Proc. IEEE Int'l. Conf. on Robotics and Automation*, Rome, Italy, 2007, pp. 391-396.

- R. Murrieta-Cid, L. Munoz, M. Alencastre, A. Sarmiento, S. Kloder, S. Hutchinson, F. Lamiraux and J.P. Laumond, "Maintaining Visibility of a Moving Holonomic Target at a Fixed Distance with a Non-Holonomic Robot," *Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems*, Edmonton, Canada, 2005, pp. 2028-2034.

- S. Kloder and S. Hutchinson, "Path Planning for Permutation-Invariant Multi-Robot Formations," *Proc. IEEE Int'l. Conf. on Robotics and Automation*, Barcelona, Spain, 2005, pp. 1797-1802.

- S. Kloder, S., S. Bhattacharya and S. Hutchinson, "A configuration space for permutation-invariant multi-robot formations," *Proc. IEEE Int'l. Conf. on Robotics and Automation*, New Orleans, LA, 2004, pp. 2746- 2751.